PEARSON EDEXCEL INTERNATIONAL AS LEVEL

INFORMATION TECHNOLOGY

Student Book 1

Sarah Lawrey
Alan Jarvis

Published by Pearson Education Limited, 80 Strand, London, WC2R 0RL
https://www.pearson.com/international-schools

Copies of official specifications for all Edexcel qualifications may be found on the website: https://qualifications.pearson.com

Text © Pearson Education Limited 2024
Development edited by Martin Payne
Copy edited by Linda Mellor
Proofread by Sakshi Agarwal, Tracey Cowell, Enya Holland, Sophie Robinson
Indexed by Pam Scholefield
Designed by EMC
Typeset by Straive Ltd
Original illustrations © Pearson Education Limited 2024
Cover design by Pearson Education Ltd 2024

Inside front cover: **Shutterstock.com**/Dmitry Lobanov

The right of Sarah Lawrey and Alan Jarvis to be identified as authors of this work has been asserted by them in accordance with the Copyright, Designs and Patents Act 1988.

First published 2024
27 26 25 24
10 9 8 7 6 5 4 3 2 1

British Library Cataloguing in Publication Data
A catalogue record for this book is available from the British Library

ISBN 978 1 292 73612 9

Copyright notice
All rights reserved. No part of this publication may be reproduced in any form or by any means (including photocopying or storing it in any medium by electronic means and whether or not transiently or incidentally to some other use of this publication) without the written permission of the copyright owner, except in accordance with the provisions of the Copyright, Designs and Patents Act 1988 or under the terms of a licence issued by the Copyright Licensing Agency, 5th Floor, Shackleton House, 4 Battlebridge Lane, London, SE1 2HX (www.cla.co.uk). Applications for the copyright owner's written permission should be addressed to the publisher.

Printed in the UK by Bell & Bain Ltd, Glasgow

Acknowledgements
See page XX

Endorsement statement
In order to ensure that this resource offers high-quality support for the associated Pearson qualification, it has been through a review process by the awarding body. This process confirms that this resource fully covers the teaching and learning content of the specification or part of a specification at which it is aimed. It also confirms that it demonstrates an appropriate balance between the development of subject skills, knowledge and understanding, in addition to preparation for assessment.

Endorsement does not cover any guidance on assessment activities or processes (e.g. practice questions or advice on how to answer assessment questions), included in the resource nor does it prescribe any particular approach to the teaching or delivery of a related course.

While the publishers have made every attempt to ensure that advice on the qualification and its assessment is accurate, the official specification and associated assessment guidance materials are the only authoritative source of information and should always be referred to for definitive guidance.

Pearson examiners have not contributed to any sections in this resource relevant to examination papers for which they have responsibility.

Examiners will not use endorsed resources as a source of material for any assessment set by Pearson.

Endorsement of a resource does not mean that the resource is required to achieve this Pearson qualification, nor does it mean that it is the only suitable material available to support the qualification, and any resource lists produced by the awarding body shall include this and other appropriate resources.

COURSE STRUCTURE IV
ABOUT THIS BOOK V
ASSESSMENT OVERVIEW VII
SPECIFICATION COVERAGE IX
INTRODUCTION XIII
UNIT 1 1
UNIT 2 128
GLOSSARY 254
INDEX 261

UNIT 1 1

TOPIC 1 HARDWARE AND SOFTWARE 2
TOPIC 2 NETWORKS 31
TOPIC 3 THE ONLINE ENVIRONMENT 63
TOPIC 4 IT SYSTEMS 79
TOPIC 5 DATA AND DATABASES 92
TOPIC 6 WIDER ISSUES 114

UNIT 2 128

TOPIC 7 UNDERSTANDING THE FUNCTIONALITY OF HTML 129
TOPIC 8 UNDERSTANDING THE FUNCTIONALITY OF CSS 157
TOPIC 9 UNDERSTANDING THE FUNCTIONS OF JAVASCRIPT 194
TOPIC 10 DESIGNING WEB PAGES 221
TOPIC 11 THE SEMANTIC WEB 242

GLOSSARY 254

INDEX 261

ABOUT THIS BOOK

This book is designed for students following the Edexcel International Advanced Level (IAL) IT specification. It covers the first year of the international A level qualification (AS level).

The book has been carefully structured to enable students to develop the knowledge and skills set out in the specification. It is organised into two units. Each of the units is split into several topics. Each topic begins by listing the key learning objectives and includes a getting started activity to introduce the concepts. There is a mix of learning points, key terms and activities throughout. Checkpoint questions and exam practice questions at the end of topics help assess understanding of the key learning objectives. Answers for the practice questions can be found online.

The content for Unit 1 is assessed in a two-hour written exam with questions which can include multiple choice and a range of open response questions with short and extended answers. The Unit 2 content is assessed with a three-hour written exam consisting of both theoretical and practical questions. Some responses will be written and some will involve submitting files. Students will be assessed according to four assessment objectives as set out below.

Demonstrate knowledge and understanding of the concepts of information technology.

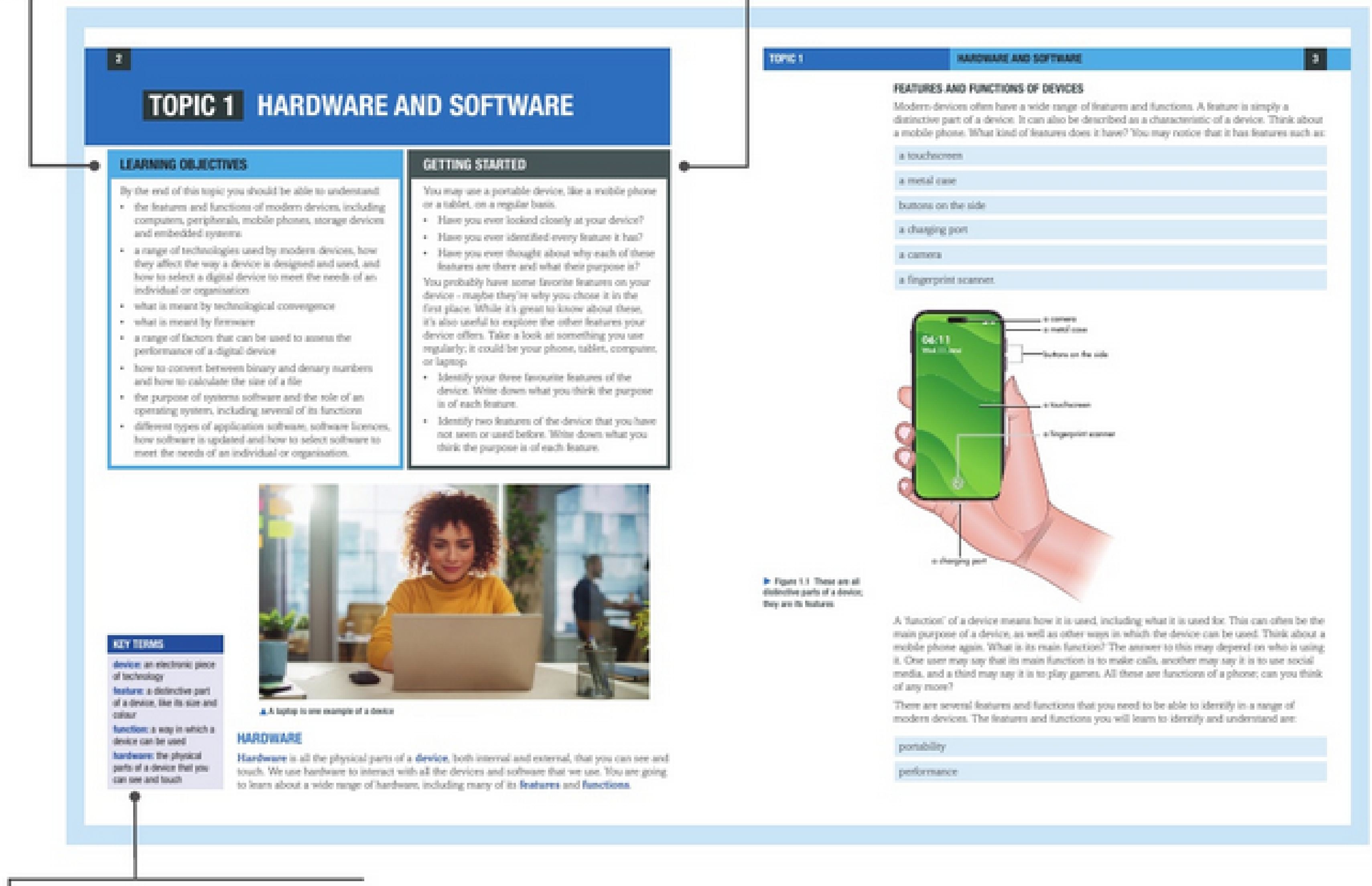
Apply knowledge and understanding of the concepts of information technology.

Analyse and evaluate IT problems.

Use analysis and evaluation to design solutions.

Learning objectives
Each topic starts with a list of key learning objectives.

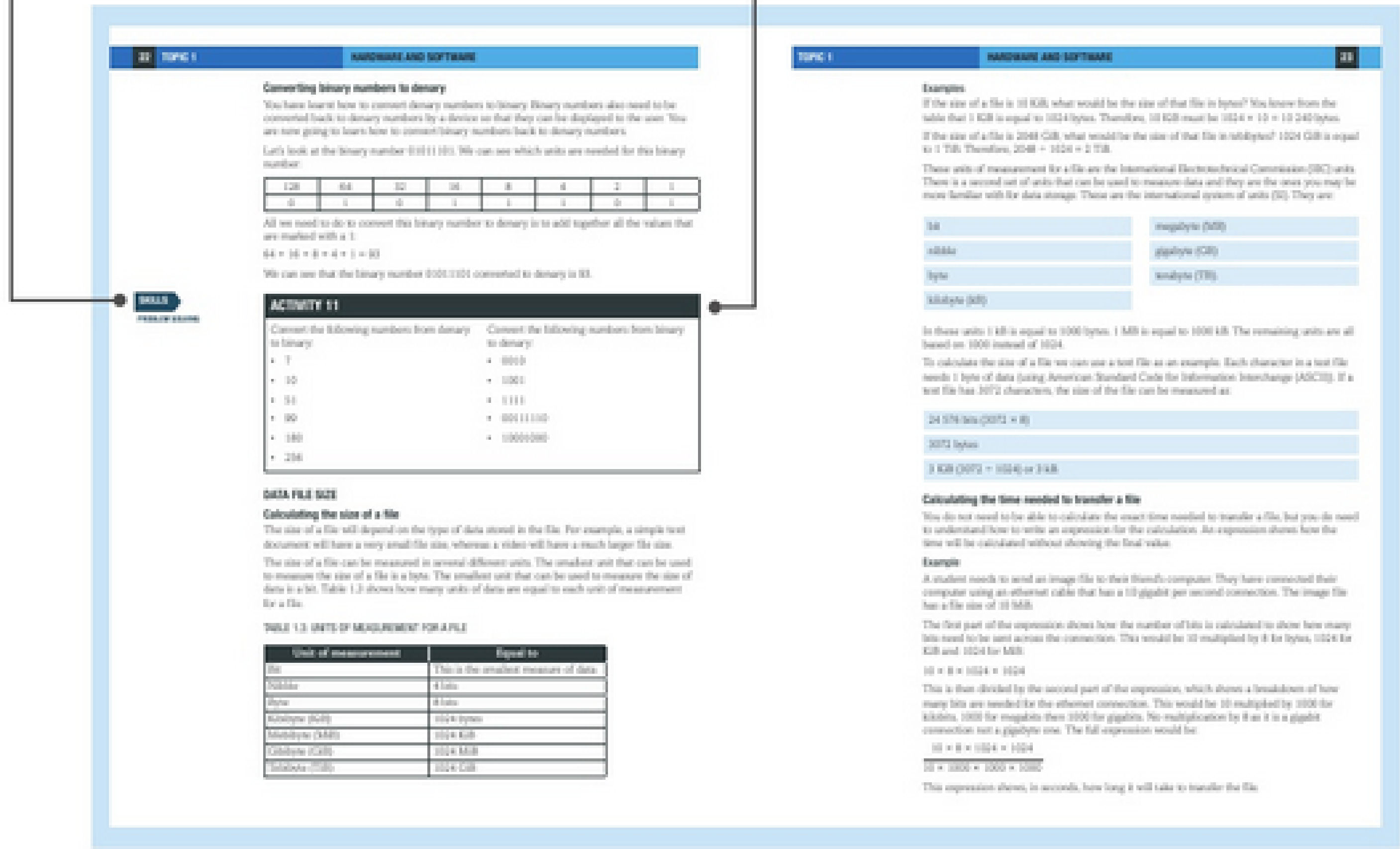
Getting started
Questions are designed to stimulate discussion and use of prior knowledge. These can be tackled as individuals, pairs, groups or by the whole class.



Key terms
Key terms are explained as they appear in the text.

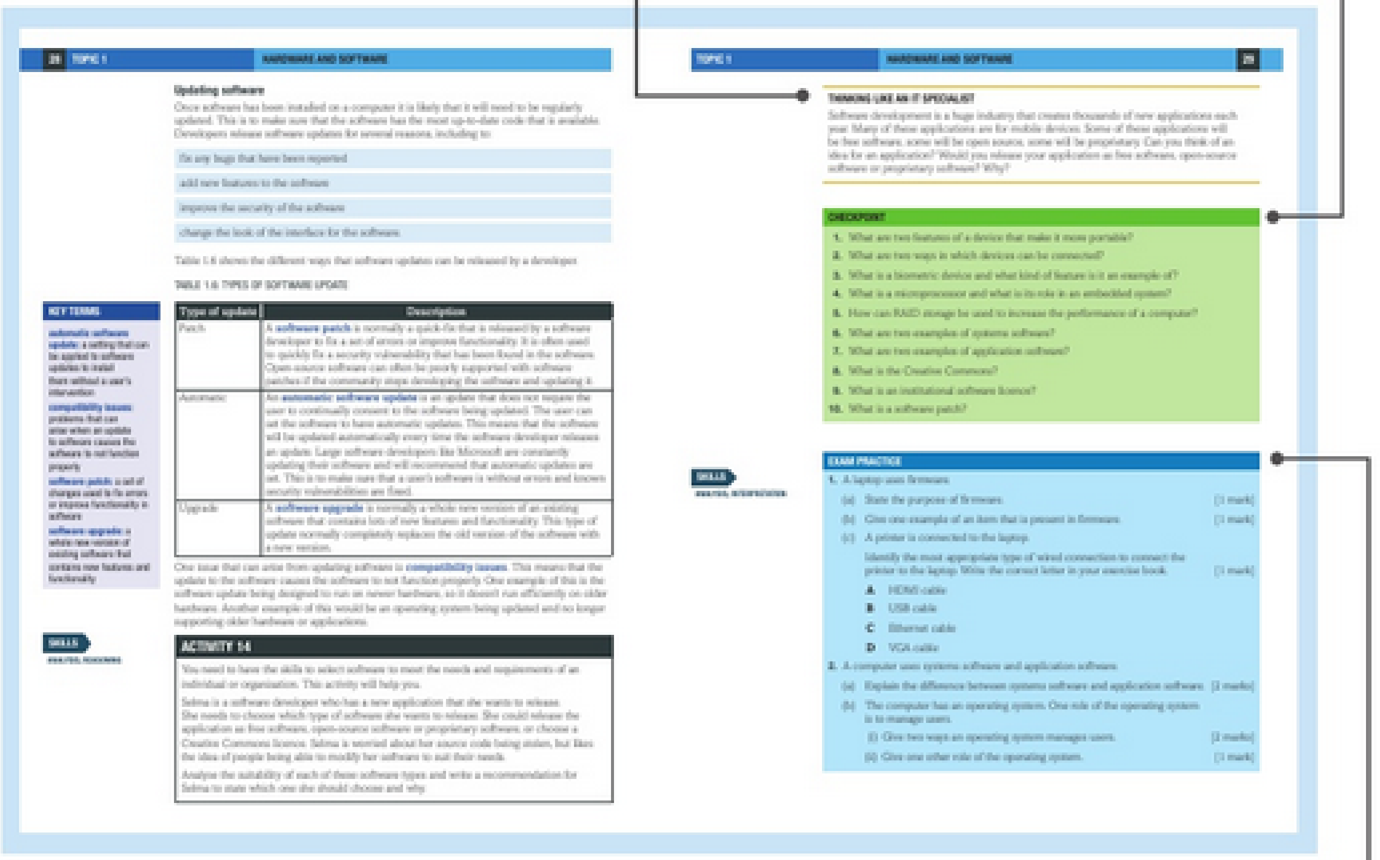
Skills
Activities have been assigned key skills, allowing for a strong focus on particular academic qualities. These transferable skills are highly valued in further study and the workplace.

Activity
Each topic includes activities to embed understanding.



Thinking like an IT specialist
These sections provide opportunity to explore an aspect of IT in more detail to deepen understanding.

Checkpoint
Questions to check understanding of the key learning points in each chapter. These are NOT exam-style questions.



Exam practice
Exam-style questions are found at the end of each topic. They are tailored to the Pearson Edexcel specification.

ASSESSMENT OVERVIEW

The following tables give an overview of the assessment for this course. You should study this information closely to help ensure that you are fully prepared for this course and know exactly what to expect in each part of the assessment.

Content and assessment overview

IAS Unit 1	*Unit code: WIT11/01	
Externally assessed Written examination: 2 hours Availability: June First assessment: June 2019 80 marks	50% of the total IAS	25% of the total IAL
Content overview Students must study **all** of the following topics: • Topic 1: Hardware and software • Topic 2: Networks • Topic 3: The online environment • Topic 4: IT systems • Topic 5: Data and databases • Topic 6: Wider issues. Students will: • demonstrate knowledge and understanding of the concepts of information technology • apply knowledge and understanding of the concepts of information technology • analyse and evaluate information technology problems • use analysis and evaluation of design solutions.		
Assessment overview • This paper may include multiple-choice, short open-response, open-response and extended open-response questions. • All four assessment objectives are assessed. • Candidates must answer all questions.		

<table>
<tr><th>IAS
Unit 2</th><th colspan="2">*Unit code:
WIT12/01</th></tr>
<tr><td>Externally assessed practical examination
Written examination: 3 hours
Availability: June
First assessment: June 2019
80 marks</td><td>50% of the total IAS</td><td>25% of the total IAL</td></tr>
<tr><td colspan="3">Content overview
Students must study all of the following topics:
• Topic 7: Understanding the functionality of HTML
• Topic 8: Understanding the functionality of CSS
• Topic 9: Understanding the functions of Javascript
• Topic 10: Designing web pages
• Topic 11: The semantic web.
Students will:
• demonstrate knowledge and understanding of the concepts of information technology
• apply knowledge and understanding of the concepts of information technology
• analyse and evaluate information technology problems
• use analysis and evaluation of design solutions.</td></tr>
<tr><td colspan="3">Assessment overview
• This paper consists of theoretical and practical questions and may include short open-response, open-response and extended open-response questions.
• Data files will be provided for some questions. Some responses will be written, others will involve the submission of files.
• All four assessment objectives are assessed.
• Each candidate will need access to a computer during the examination.
• Candidates must answer all questions.</td></tr>
</table>

SPECIFICATION COVERAGE

UNIT 1

TOPIC 1: HARDWARE AND SOFTWARE	SPECIFICATION NUMBER	KEY IDEAS
1.1 Hardware	1.1.1; 1.1.4	• The features and functions of modern devices, including computers, peripherals, mobile phones, storage devices and embedded systems.
	1.1.2; 1.1.9; 1.2.6	• The range of technologies used by modern devices, how they affect the way a device is designed and used, and how to select a digital device to meet the needs of an individual or organisation.
	1.1.3	• What is meant by technological convergence.
	1.1.5	• What is meant by firmware.
	1.1.6	• The range of factors that can be used to assess the performance of a digital device.
	1.1.7, 1.1.8	• How to convert between binary and denary numbers and how to calculate the size of a file.
1.2 Software	1.2.1; 1.2.2	• The purpose of systems software and the role of an operating system, including several of its functions.
	1.2.3; 1.2.4. 1.2.5	• Different types of application software, software licences, how software is updated and how to select software to meet the needs of an individual or organisation.

TOPIC 2: NETWORKS	SPECIFICATION NUMBER	KEY IDEAS
2.1 Network models and protocols	2.2.3	• The role of a range of different network components.
	2.2.1	• How different network transmission media are used and how they function.
	2.1.1	• The features and functions of different network models.
	2.2.2	• A range of ways the performance of a network can be measured.
	2.1.2	• The purpose and features of different communication methods, like ethernet, Wi-Fi, Bluetooth®, cellular, infrared and ZigBee®.
	2.1.3	• Understand a range of network standards and protocols, including their purpose and how they function, and the role of an IP address and a MAC address in a network.
2.2 Network design and implementation	2.2.4	• How to design a network to meet a set of requirements, including the location of devices.
2.3 Network security	2.3.1; 2.3.2	• Why a network needs to be kept secure and how this can be done using hardware and software security solutions.

TOPIC 3: THE ONLINE ENVIRONMENT	SPECIFICATION NUMBER	KEY IDEAS
3.1 The internet and the world wide web	3.1.1; 3.1.2	• The features and functions of the internet and the world wide web.
	3.1.3	• What is meant by static and dynamic web page content and how they are different.
	3.1.4; 3.1.5	• The role of client-side and server-side scripting.
3.2 Operating online	3.2.1; 3.2.2; 3.2.3	• The impact of working in online environments.
3.3 Online communities	3.3.1; 3.3.2; 3.3.3	• The concept of an online community and the impact they have on an individual and an organisation, including a range of ways they can be used to make money.
	3.2.2	• The security risks of storing personal data online and how it can be protected.
3.4 The cloud	3.4.1; 3.4.2	• The concept and impact of cloud storage and cloud computing.
	3.2.3	• What is meant by a digital footprint and the positive and negative effects of this.

TOPIC 4: IT SYSTEMS	SPECIFICATION NUMBER	KEY IDEAS
4.1 Systems design	4.1.1	• The main components of an IT system.
	4.1.2	• How to break a system down into smaller parts.
4.2 Dataflow	4.2.1	• What dataflow diagrams are used for and how to create them.
4.3 Flowcharts	4.3.1; 4.3.2	• What flowcharts are used for and how to create them.
	4.1.3	• How to design and evaluate part of an IT system and a whole IT system.
4.4 Systems	4.1.3; 4.1.4	• The advantages and disadvantages of a given IT system for an individual and an organisation, including the use of digital, peripheral and storage devices.

TOPIC 5: DATA AND DATABASES	SPECIFICATION NUMBER	KEY IDEAS
5.1 Data and information	5.1.1	• What is meant by data and how it is different to information.
	5.1.2	• What is meant by structured and unstructured data.
	5.1.3	• The benefits of data extraction to create meaningful data for organisations.
5.2 Structured data	5.2.1; 5.2.2	• The different components of a database including relational databases and the different types of keys.
	5.2.3; 5.2.4	• The different types of relationships that can be created between tables in a database.
5.3 Structured query language (SQL)	5.3.1; 5.3.2	• How to create queries using structured query language (SQL).

TOPIC 6: WIDER ISSUES	SPECIFICATION NUMBER	KEY IDEAS
6.1 Environmental	6.1.1	• The environmental impact of creating, using and disposing of technological equipment.
	6.1.2	• The positive impact of using technology on the environment when used for monitoring and using resources efficiently.
6.2 Legal, moral and ethical	6.2.1; 6.2.2	• The legal, moral and ethical issues connected to the use of ICT systems.
6.3 Society	6.3.1	• The impact of wireless access being available everywhere.

UNIT 2

TOPIC 7: UNDERSTANDING THE FUNCTIONALITY OF HTML	SPECIFICATION NUMBER	KEY IDEAS
7.1 Document structure	7.1.1; 7.1.2; 7.1.3; 7.1.4; 7.1.5	• How HTML is used to structure web pages.
7.2 Structural markup	7.2.1; 7.2.2; 7.2.3; 7.2.4; 7.2.5	• What semantic markup is and how to do it.
7.3 Page components	7.3.1; 7.3.2; 7.3.3; 7.3.4; 7.3.5; 7.3.6	• How to include images, tables, forms and media in a web page.

TOPIC 8: UNDERSTANDING THE FUNCTIONALITY OF CSS	SPECIFICATION NUMBER	KEY IDEAS
8.1 Writing CSS	8.1.1; 8.1.2; 8.1.3; 8.1.4; 8.1.5; 8.1.6; 8.1.7	• How CSS can be used within an HTML document.
8.2 Styling web pages	8.2.1; 8.2.2; 8.2.3; 8.2.4; 8.2.5	• How to apply styles to a range of HTML elements.
8.3 Positioning content	8.3.1; 8.3.2; 8.3.4	• How to position content on a web page.
8.4 CSS animations	8.4.1	• How to create responsive web pages that can be used with different size screens.

TOPIC 9: UNDERSTANDING THE FUNCTIONS OF JAVASCRIPT	SPECIFICATION NUMBER	KEY IDEAS
9.1 Document Object Model	9.1.1	• How JavaScript uses the Document Object Model to interact with HTML.
9.2 Regular expressions	9.2.1; 9.2.2; 9.2.3	• How to use regular expressions to validate user input.
9.3 Programmed functionality	9.3.1; 9.3.2; 9.3.3	• How to use programming functionality in JavaScript.
	9.3.4	• How to create web page features using HTML, CSS and JavaScript.

TOPIC 10: DESIGNING WEB PAGES	SPECIFICATION NUMBER	KEY IDEAS
10.1 Principles of design	10.1.1	• How design tools such as wireframes, mock-ups, style guides and prototyping are used to design web pages.
	10.1.2	• What web-safe fonts are.
	10.1.3	• How to use design principles to create effective page layouts and designs.
10.2 Principles of accessibility	10.2.1	• What we can do to make web pages accessible to everyone.
	10.3.1	• How to design helpful navigation systems.
10.3 Principles of usability	10.3.2	• Targeting specific devices and web browsers.

TOPIC 11: THE SEMANTIC WEB	SPECIFICATION NUMBER	KEY IDEAS
11.1 Writing semantic code	11.1.1	• What is meant by semantic markup on a web page.
	11.1.4	• How semantic markup is used by web browsers and search engines.
	11.1.3; 11.1.6	• How to add semantic markup to add structural meaning.
	11.1.5	• How to add semantic markup to add textual meaning.
	11.1.7	• How the Web Accessibility Initiative – Accessible Rich Internet Applications (WAI-ARIA) can be used to provide additional semantics for accessibility.

INTRODUCTION

Welcome to your International A Level course in Information Technology (IT).

IT is a complex and interesting subject to study at this level. During your studies you will learn about new technologies, the impact of technologies and what the future may hold for technology. You will develop a range of new skills to analyse technology and to build systems, such as web pages. If you have studied ICT at iGCSE level, you will be able to bring forward and enhance the skills learned there.

Technology is a constantly developing element of society and many industries are now dependent on it, especially for communication. Whatever you choose to study after your International A Levels or select as a career, it will likely use some aspect of IT. Doctors, veterinarians, architects, web developers, entrepreneurs, accountants, analysts, graphic designers and engineers are just some of the careers that use a wide range of technology. People working in these industries need to be able to analyse and use technologies to get the best out of their career opportunities. The knowledge and skills that you learn during this course will help to equip you in some respect for whatever awaits you in your future life.

You have chosen an interesting, engaging and highly relevant course of study. With hard work and determination you will enjoy and learn with excellence during this IT course.

Have fun!

Sarah

TOPIC 1
HARDWARE AND SOFTWARE

Modern devices can have a lot of features and functions, and throughout this topic you will learn about these. These features can be hardware- or software-based. We need to understand these features so as to select the most suitable device for ourselves and others. A hardware feature like the size of the screen could be important for our use, as could a software feature like the operating system that the device uses. This topic will provide you with information about how to assess the performance of a device and help you understand which factors contribute to the performance of a device. You will also learn some mathematical aspects of IT, such as how to convert between binary and denary numbers and how to calculate the size of a file.

TOPIC 1 HARDWARE AND SOFTWARE

LEARNING OBJECTIVES

By the end of this topic you should be able to understand:

- the features and functions of modern devices, including computers, peripherals, mobile phones, storage devices and embedded systems
- a range of technologies used by modern devices, how they affect the way a device is designed and used, and how to select a digital device to meet the needs of an individual or organisation
- what is meant by technological convergence
- what is meant by firmware
- a range of factors that can be used to assess the performance of a digital device
- how to convert between binary and denary numbers and how to calculate the size of a file
- the purpose of systems software and the role of an operating system, including several of its functions
- different types of application software, software licences, how software is updated and how to select software to meet the needs of an individual or organisation.

GETTING STARTED

You may use a portable device, like a mobile phone or a tablet, on a regular basis.

- Have you ever looked closely at your device?
- Have you ever identified every feature it has?
- Have you ever thought about why each of these features are there and what their purpose is?

You probably have some favorite features on your device - maybe they're why you chose it in the first place. While it's great to know about these, it's also useful to explore the other features your device offers. Take a look at something you use regularly; it could be your phone, tablet, computer, or laptop.

- Identify your three favourite features of the device. Write down what you think the purpose is of each feature.
- Identify two features of the device that you have not seen or used before. Write down what you think the purpose is of each feature.

▲ A laptop is one example of a device

HARDWARE

Hardware is all the physical parts of a **device**, both internal and external, that you can see and touch. We use hardware to interact with all the devices and software that we use. You are going to learn about a wide range of hardware, including many of its **features** and **functions**.

KEY TERMS

device: an electronic piece of technology

feature: a distinctive part of a device, like its size and colour

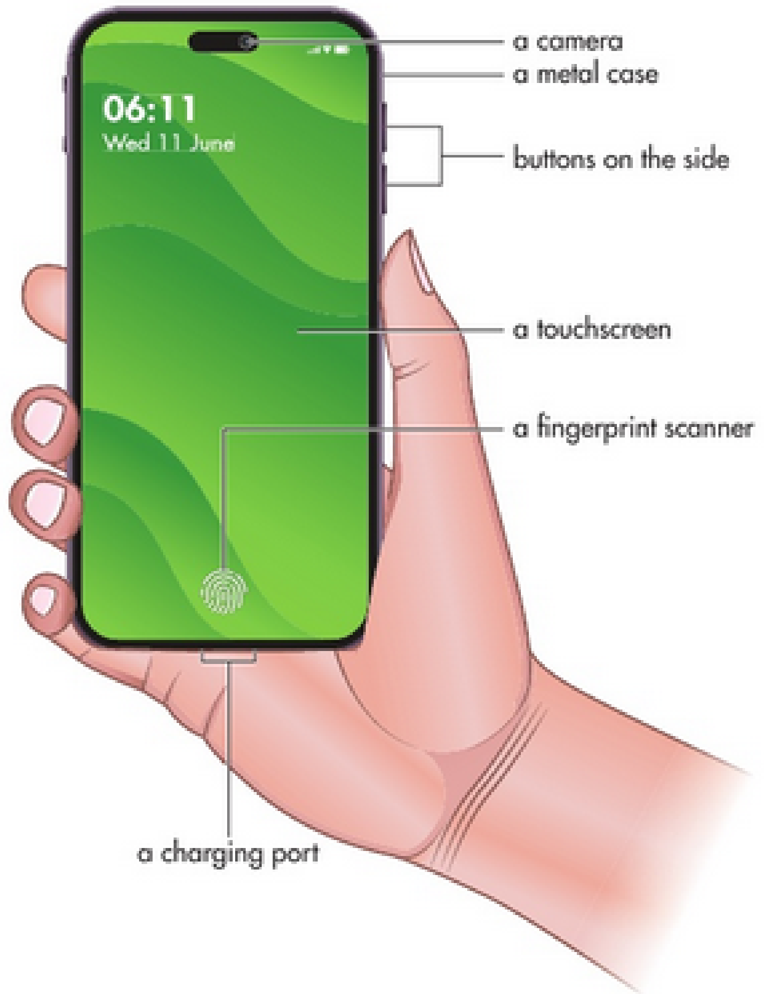
function: a way in which a device can be used

hardware: the physical parts of a device that you can see and touch

FEATURES AND FUNCTIONS OF DEVICES

Modern devices often have a wide range of features and functions. A feature is simply a distinctive part of a device. It can also be described as a characteristic of a device. Think about a mobile phone. What kind of features does it have? You may notice that it has features such as:

- a touchscreen
- a metal case
- buttons on the side
- a charging port
- a camera
- a fingerprint scanner.



▶ Figure 1.1 These are all distinctive parts of a device; they are its features

A 'function' of a device means how it is used, including what it is used for. This can often be the main purpose of a device, as well as other ways in which the device can be used. Think about a mobile phone again. What is its main function? The answer to this may depend on who is using it. One user may say that its main function is to make calls, another may say it is to use social media, and a third may say it is to play games. All these are functions of a phone; can you think of any more?

There are several features and functions that you need to be able to identify in a range of modern devices. The features and functions you will learn to identify and understand are:

- portability
- performance

- storage
- user interface
- connectivity
- media support
- energy consumption
- expansion capability
- security features.

To understand how these features and functions are used in modern devices, you will learn about how they are part of a range of modern devices, such as:

- computers
- peripheral devices
- mobile phones
- storage devices
- embedded systems.

Portability

KEY TERM

portability: how easy it is to carry a device around from one place to another

The **portability** of a device is how easy it is to carry around. Often, the most portable device is the smallest device because it is easier to fit into your pocket or bag. This is why technology companies often spend a lot of money trying to make their devices smaller in size, as this makes them more portable. However, there can be drawbacks to making a device as small as possible.

SKILLS

ANALYSIS, COLLABORATION

ACTIVITY 1

A technology company wants to manufacture a new laptop. They are thinking about the design of their laptop. You are part of their design team, and they have two questions they need you to consider:

- What would make the laptop more portable?
- What drawbacks could there be from making the laptop very small?

Write your thoughts for each of the questions. Share your thoughts with a partner and have a discussion to reflect on what you have written. In your discussion you could consider these questions:

- Do you have similar or different ideas about how to make the laptop more portable?
- What is the best idea for making the laptop more portable?
- Do you have similar or different ideas about what the drawbacks might be?
- What do you think is the most important drawback for the technology company to consider?

Portability can be very important for some devices, but not as important for others. It is important for a device that will regularly be carried around to be portable. This type of device could be a mobile phone, tablet or an external storage device. It is not as important for a device that will be stationary most of the time to be portable. This type of device could be a desktop computer, a games console or a smart TV.

Think about the following list of devices. Is it important for each of them to be portable? Why?

- Keyboard
- Mouse
- SD card
- Webcam
- Speaker
- Printer

Performance

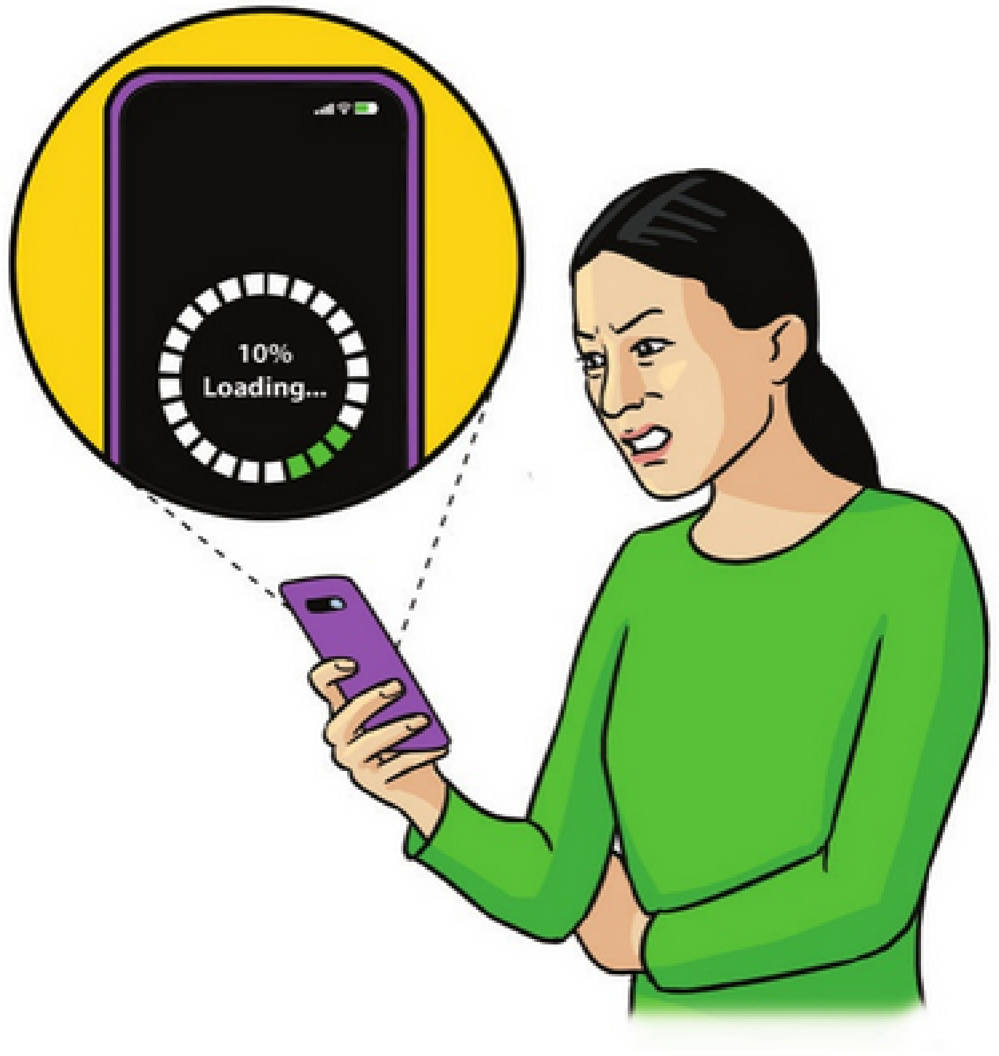
KEY TERMS

application: software that is designed to perform a specific task

performance: how well a device performs its intended purpose

You may have tried to use an **application** on your mobile phone or computer and found yourself frustrated about the time it is taking to open. This frustration may well be due to the **performance** of the device, which is simply how well it works for its intended function. One of the functions of a computer is to open applications. If a computer opens an application quickly, this means the performance is good. If a computer opens an application slowly, it has poor performance and will most likely make a user frustrated.

Performance is more important for some devices than others, depending on how it is intended to be used.



▶ Figure 1.2 If the performance of a device is poor, it can lead to applications opening slowly

SKILLS

CRITICAL THINKING, COLLABORATION

ACTIVITY 2

A train station has a system that includes several different devices. It has:

- a desktop PC that receives all the **data** about the different trains that are using the station
- a speaker that is used to make announcements to customers
- a monitor that allows customers to see when trains will be arriving and departing.

Find two other people to work with to create a group. In your group discuss your thoughts about the following statements.

- It is important that the PC has a high level of performance.
- It is not important that the speaker has a high level of performance.
- It is not important that the monitor has a high level of performance.

Do you agree with these statements? Why?

Storage

Storage is a key part of many different devices. It can also be a device on its own. Storage can be found in devices such as desktop PCs, laptops, tablets, mobile phones and games consoles. Examples of storage devices are a hard disk drive (HDD), solid-state drive (SSD) and an optical disk drive. Considering the type and size of storage can be very important.

Think about a tablet device. A user will most likely want to store a lot of data on the tablet. They may have lots of photos, videos and movies that they want to store. This means that it will need a large storage capacity. However, if a larger amount of storage is built into the tablet, for example 2 terabytes (TB), this will increase the size and weight of the tablet. This may reduce the portability of it and make it more difficult for the user to carry around. The type of storage built into a device is also key. Have you ever dropped your phone or tablet? You probably dread doing this! If a tablet contained **magnetic storage**, the storage would easily be damaged if the device was dropped. This is because magnetic storage has moving parts. **Solid-state storage** is used in tablets because it does not have any moving parts and, so, it is less likely to break if dropped.

Solid-state storage uses electronics (integrated circuit devices) to store data. This means it can be made much smaller and much faster (there are no motors needed to move parts). Solid-state storage is used in mobile phones, memory sticks and portable disc drives (SSDs). Most modern laptops also use solid-state storage because it is fast, small and light. It is also becoming more common in desktop computers because of its superior speed. An SSD up to 1 TB capacity costs about the same as an HDD with the same capacity, but higher capacity SSDs are currently much more expensive than the equivalent HDDs. Many desktop computers will have both – an SSD for the operating system and application software, and an HDD for storing large amounts of data.

There are some devices or systems that need more complex types of storage, for example redundant array of independent disks (RAID) storage. RAID storage is when multiple storage devices (mostly HDDs or SSDs) are combined to act like they are a single larger storage device.

It is often very expensive to buy storage devices with a large capacity. However, computers that perform intensive tasks like **data mining** may need very large amounts of storage to have a high level of performance, because they need to store huge amounts of data, read a lot of data and write a lot of data.

RAID storage could be used in a computer performing this type of task, since it may be cheaper to buy multiple smaller storage devices and combine them.

KEY TERMS

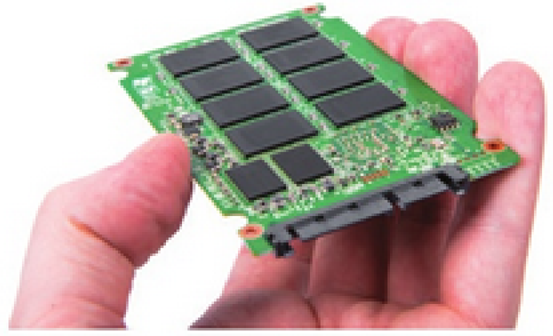
data: text, values, symbols, images and sound in their raw form

data mining: the process of analysing large databases to find patterns and trends

magnetic storage: a type of storage device that uses electromagnets to store data

solid-state storage: a type of storage device that stores data on silicon chips

storage: a piece of hardware that is used to store data



▲ Solid-state storage

KEY TERM

supercomputer: a computer with a very high level of performance

A **supercomputer** is a powerful computer that will probably use RAID storage. This type of RAID storage is called RAID 0 and it is used for increasing the performance of computers.

There is also RAID 1 storage. RAID 1 storage is used for protecting data. The data is stored on one storage device in the RAID setup. All other storage devices in the RAID setup are then set to be a mirror of the first storage device so that they store the same data that is stored on the first device. This is like having a live backup of the data. If one of the storage devices in the RAID starts to fail, it is easy to remove it and replace it with a new one without losing any data.

▶ An array of disks used for RAID storage

SKILLS

CREATIVITY

ACTIVITY 3

Imagine you are a designer at a technology company. You have been asked by your manager to design a new games console. You will need to think about the following.

- Will the device be a standard tabletop games console or a hand-held games console?
- Is portability important? Why?
- Is performance important? Why?
- Is storage capacity important? Why?

In your exercise book, start creating a design brief for the games console by writing down your thoughts about these questions. You will be adding to this design brief in other activities to consider other features of the device.

KEY TERM

user interface: the part of a device that a person uses to interact with the device

User interface

Many devices have some type of **user interface**. This is normally what a person uses to interact with the device. A user interface could be as simple as a small number of buttons, or it could be as complex as a range of icons and other means of communicating with the device, for example a text entry interface. This will depend on the device. A mouse has a simple hardware-based user interface – usually with two buttons. A desktop PC has a more complex user interface – it is graphical and often uses windows, icons, menus and pointers.

▶ Designing a user interface

The user interface of a device is important: it needs to be as simple as possible so that a person is able to easily use the device. This might be easy when the device consists of only two buttons, but it becomes more difficult when the device needs to be used for many different tasks, such as writing documents, playing music and editing photographs.

SKILLS

INTERPRETATION

ACTIVITY 4

Copy this table into your exercise book. Write a description of what an interface for each device would include. The first two have been done for you.

Device	User interface description
Mouse	Normally has two buttons, a left and a right button. Can sometimes have a scroll wheel.
Desktop PC	Has a graphical user interface that often includes windows, icons, menus and pointers.
Keyboard	
Speaker	
Mobile phone	
Printer	
Smart watch	
Smart TV	

KEY TERMS

connectivity: the ability to connect devices to other devices

wired connection: a method of connecting devices using cables

wireless connection: a method of connecting devices using radio waves

Connectivity

Our modern world is a very connected place and a central part of this is the number of devices that are connected. Can you think of a device that you use that is not connected to another device in some way? For this reason the **connectivity** of a device is essential. Devices are normally connected either by a **wired** or **wireless connection**. For a wired connection, devices need to have a **port** for the cable. The port will be for a specific type of cable only. The types of cables that can be connected to devices include:

universal serial bus (USB) cables

KEY TERMS

Bluetooth chip: a piece of hardware in a device that allows it to have a Bluetooth connection

port: a connection point in a device that a cable is plugged into

wireless network interface card (WNIC): a piece of hardware in a device that allows it to connect to a wireless network

high-definition multimedia interface (HDMI) cables

video graphics array (VGA) cables

display port (DP) cables.

A wireless connection does not require a device to have a physical connection port. It does however require that the device has a **wireless network interface card** (WNIC) or **Bluetooth chip** built into it. Another type of wireless connection is near field communication (NFC). This is a secure method of wireless encryption that can be used to allow two devices to communicate when they are in very close proximity to each other.

The type of connection port or wireless connection hardware that a device needs can affect the size of the device. You may have thought before that it would be helpful if your portable device (e.g. mobile phone or tablet) had an HDMI port. This would mean that you could easily connect it to most smart TVs to view photos and videos from your phone on a big screen. However, your device would then need to be bigger to include this type of port, affecting its portability.

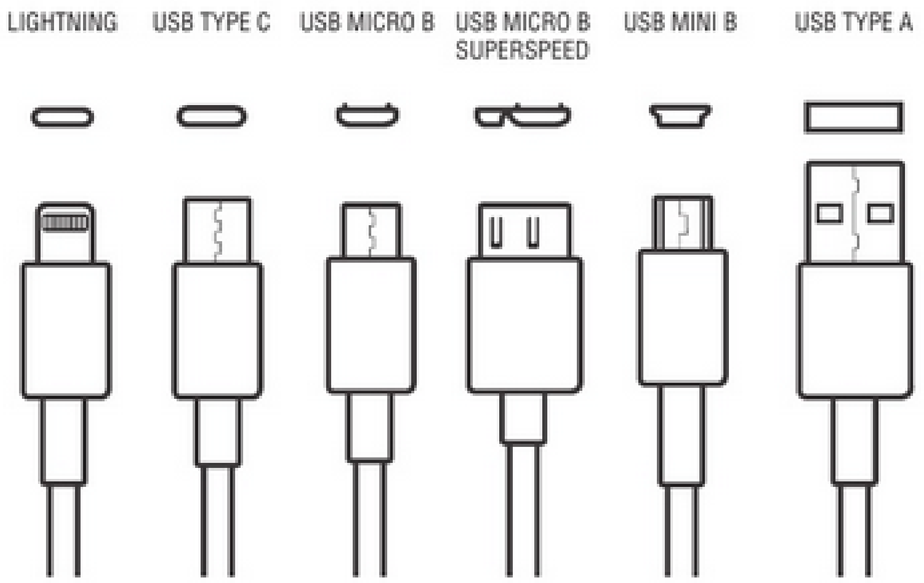
Different types of cables have different speeds for the transmission of data. This means that the type of connection a device uses can affect the performance of the device. If the cable does not transmit data to the device at a fast enough speed, this may affect its performance. For example, a person might use their computer to watch high-definition movies. They could choose to use a VGA cable or an HDMI cable to connect the monitor to the computer. An HDMI cable transmits data faster than a VGA cable. This means that the viewing quality of the movie will likely be higher if an HDMI cable is used.

There are now different versions of USB cables available, for example USB2, USB3 and USB-C. USB3 was developed to transmit data faster than USB2. This means that if a device is connected using a USB3 connection instead of a USB2 connection, it may improve the performance of the device. For example, a student using a USB cable to connect their external hard drive to their computer and backup their files, would find the process quicker if they chose a USB3 connection instead of a USB2.

USB-C is the most recent version of USB cables. It was developed to try and create a standard version for all USB connections. The aim was to make all USB connection cables universal for all devices, allowing them to all use the same connection that would provide the correct power and speed of transmission for each device. This meant that devices around the world would use a standard connection, so that people could share USB cables if needed.

Every USB cable is the same at one end. One end of the cable has a wide, flat connection that plugs into a computer or a power adapter. This is called a type A connection. When USB cables were first invented in the 1990s, both ends of the cable were type A connections. Now, the other end of the USB cable is different depending on the type of cable. There are several types of connection on this end, including type B, mini USB, micro USB, USB-C and Apple's lightning connection. Each connection is used for different types of device and provides a different level of power for the device, and different speeds of data transmission. For example, most modern mobile phones use a USB-C connection, apart from Apple iPhones, which use Apple's lightning connection. This is because they can charge the device and have a fast speed for data transmission. Micro USB connections are mainly used for charging devices such as power banks and also wireless headphones, as they are a faster connection for just charging a device.

▼ Figure 1.3 The different types of USB cables



KEY TERM

storage media: the type of storage that can be used for a device; for example, the device can have a memory card slot or a USB port that allows a USB drive to be connected

Media support

People want a lot of their data readily available on their devices. For example, most people now store far more than just contact numbers and texts on a phone. They might store podcasts, videos, movies, documents and hundreds or even thousands of photographs. The storage a device needs for all this data could be quite large, which would make any portable device larger and possibly heavier. Therefore, **storage media** support has become increasingly important. For example, many mobile phones now support the use of micro SD cards. These can be inserted into the device and used as additional storage space. This also means that the SD card can be removed and inserted into any other device that supports the use of an SD card; so, you can transfer files from one device to another.

Media support for larger devices is also changing. In the past, most computers supported optical disk drives such as CD drives or DVD drives. However, many modern computers no longer have an optical drive built into them. The manufacturing costs for optical media can be expensive and wasteful, especially if the media is a CD ROM that can only have data written to it once. This kind of plastic waste is being discouraged because of the impact on the environment. Optical disks are also fragile and prone to wear and tear. However, there are still companies and businesses that rely on optical media because they have a lot of archived data that was saved using this method; so, they still need devices that support this media. The lack of support for this media from computer manufacturers may cause them problems and they might be forced to change their data archiving approach.

When purchasing a device, you may want to consider if it supports the following media formats:

- memory cards e.g. SD cards
- flash memory e.g. USB drive
- optical disks e.g. CD ROM.

In the future, as there is a trend towards more data and software for devices being stored in the cloud, it may be that media support for devices becomes less important. As most data will be stored away from the device on remote servers, very little data may actually be stored on the device itself.

KEY TERM

energy: the amount of electricity used by a device

Energy consumption

It is increasingly important in our modern world to consider how much **energy** we are consuming. This means that we should all think about how much energy the devices that we use are consuming. Technology companies aim to make devices that consume much less energy than they used to. It is now important to many people when choosing a device that it has a low energy consumption. This can be for several reasons, including the following:

- they want to make sure that they are helping the planet by using devices that consume little energy,
- they do not want the annoyance of having to regularly charge a device because it has high energy consumption.

SKILLS
CREATIVITY

ACTIVITY 5

Find the design brief for the games console that you previously wrote in your exercise book. You now need to think about the following.

- Which other devices will your games console need to connect to?
- How will your games console connect to another device?
- What will the energy consumption of your device be like? Why is this important?

Add to your design brief by writing down your thoughts on these questions.

KEY TERM

expansion capacity: the ability to add parts to a device, for example adding more storage space to it

Expansion capability

Some devices have a set structure and size that means nothing can be added to them. For example, certain mobile phones cannot have any further storage added to them other than the storage that is built into them. Some devices can be expanded. This means that they can have elements added to them, such as more storage space. This is called the **expansion capacity** of a device. For example, some phones have a slot that can accept a micro SD card. This means that more storage space can be added to the device. This can be important to a user who wants to store lots of data on their phone. It may mean that they will only buy a phone that has expansion capacity.

SKILLS
INTERPRETATION, COLLABORATION

ACTIVITY 6

Think about how expansion capacity could be important for these devices:

- desktop PC
- video camera
- games console
- RAID storage
- smart TV
- network-attached storage (NAS).

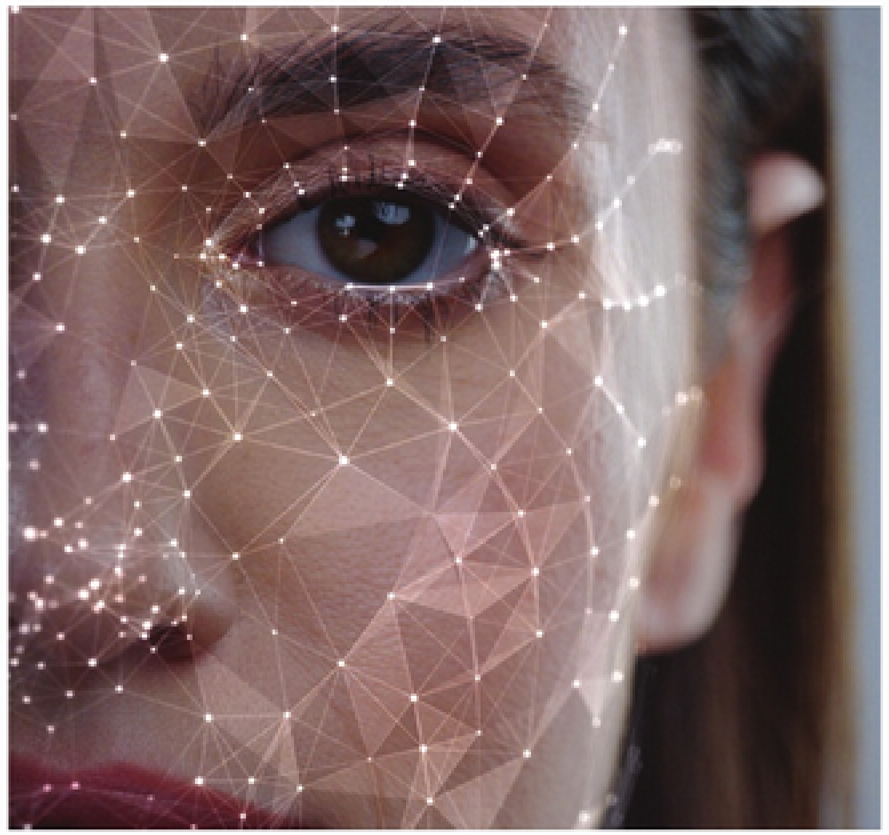
You may need to use the internet to research some of these devices to get a basic understanding of what they are. This will help you think about how they could benefit from expansion capacity.

Share your thoughts for three of these with a partner. Did you identify the same things? What did you think was the most important thing your partner identified?

KEY TERMS

biological data: data that belongs to a human, for example a fingerprint, an iris, facial features and their voice

biometric device: a type of security device that uses biological data as a form of password



▲ A facial recognition biometric device maps thousands of points on a person's face

Security features

Many people around the world now store lots of data. Much of it is personal data that people would consider confidential. Many businesses also store confidential data which is very valuable to them. It could be data about their customers or about secret products they have in development. It is essential to keep all this data safe. This means that many devices have one or more security features to make sure that any data that is stored is kept safe.

One of the most basic ways to keep data safe is by adding a password that needs to be entered to access the data. However, it has become easier for another person to guess our passwords by using special software to try thousands of different combinations until a correct password is found. **Biometric devices** were invented to help make devices more secure. A biometric device uses a type of biological password. It can use a person's **biological data** such as their fingerprint or facial features as a type of password. This type of biological data is extremely difficult to fake, which makes devices much more secure than when a standard text-based password is used.

SKILLS

CREATIVITY, COLLABORATION

ACTIVITY 7

There are other technologies that have been developed to help add security features to a device. These include:

- chip and PIN
- radio-frequency identification (RFID) tags
- NFC.

Find two other people to work with to create a group. In your group, create a short video that shows how the technologies given are used as security features.

When you have made your video, find another group who have finished and watch each other's videos. Give feedback to the other group that includes the following.

- Which security feature do you think they explain best?
- What was one thing they included that you did not include?
- What was one thing that you think they could improve?

EXAM TIP

If an exam question asks you to discuss a certain feature of a device but does not specify a device, simply think about a device that you know well. Do not get too adventurous by choosing a device that you are not as familiar with.

FEATURES THAT AFFECT THE DESIGN AND USE OF A DEVICE

You have learnt about many features and functions that are important in modern-day devices. Most of these features affect how the device is designed and used. Table 1.1 shows a range of features and a brief description about how they affect the design and use of a device. Some of these have been covered in more detail earlier in this topic.

KEY TERM

global positioning system (GPS): a satellite-based navigation system

KEY TERMS

non-volatile storage: computer storage that retains its data when the power is removed

random access memory (RAM): a type of volatile memory in a computer that stores currently running data and programs

read only memory (ROM): a type of non-volatile memory in a computer that is used to store start-up instructions for the computer

sensor: a type of input device that captures data from its immediate surroundings

touchscreen: a type of input device that allows a user to touch a screen to input data and use the screen to see the data

volatile storage: computer storage that loses its data when the power is removed

TABLE 1.1: FEATURES AND THEIR EFFECT ON THE DESIGN AND USE OF A DEVICE

Feature	Description
Global positioning system (GPS)	GPS is often included in devices that are portable. This is normally so a person can use the device as a navigation tool to get directions to where they want to go. GPS uses a network of satellites to pinpoint a location. It can also be used to find a device that has been lost or stolen by identifying the location of the device. The hardware needed for a device to use GPS is quite small – smaller than the size of a battery. This means that it can be easily built into the design of most devices if needed. GPS will increase the power usage for a device as the system can often be regularly or constantly in use by the device. GPS has many uses besides personal devices. It is used by surveyors, pilots, scientists, rescue workers and so on.
Biometrics	You have already learnt about some biometric devices. They are a helpful feature that can be built into a device to add a security feature. A biometric device can be small, for example the size of the tip of a finger. This means that it can be easily built into the design of most devices if needed.
Touchscreen	Touchscreens can be built into a range of devices, such as a mobile phone, smart watch, tablet, laptop and hand-held games console. A touchscreen can often be used in a device to save space and means the device can be made smaller. For example, if a phone needed to have both a screen and a physical keyboard built into it, the device would need to be quite large, or the screen and keyboard would need to be very small. This would make the device difficult to use. A touchscreen is a screen that can be used as a display, but also as an input device to allow data to be entered into the device. This means that both the keyboard and the screen can be combined into one, so that the display screen part of the touchscreen can be large to improve the viewing quality for the user.
Sensor	A sensor is a type of input device that collects data from its immediate surroundings. Sensors are used in a huge range of devices, from tablet computers to washing machines. Here are some examples of sensors that can be included in devices. • Light sensor – to detect the ambient light of the room to automatically adjust the brightness of the screen. Light sensors can also be used to measure the heart rate of a user in a device like a smart watch. • Accelerometer – to change the orientation of the screen depending on which way a device is held. • Motion sensor – to measure the number of steps taken by a user in a day or to detect the motion of a user who is playing a computer game. • Temperature sensor – to measure the temperature of the water that has been added to the washing machine.
Memory	Memory in a device can be **read only memory (ROM)** or **random access memory (RAM)**. ROM stores the start-up instructions for a device. It is **non-volatile storage**. RAM stores the data and instructions that are currently being processed by a device. It is **volatile storage**. The size of memory, especially RAM, can affect the performance of a device. If a device has a limited amount of RAM, its performance may not be as good. So why not build a large amount of memory into a device? Memory hardware can take up quite a lot of space! This would mean that a device needs to be bigger to have more memory built in. This could make the device less portable.
Storage	You learnt about storage earlier in the topic. The same thoughts that you have just learnt about memory can be applied to storage. If a device has more storage, it can increase the performance of the device. However, storage hardware can also take up quite a lot of space in a device.

(Continued)

TABLE 1.1: FEATURES AND THEIR EFFECT ON THE DESIGN AND USE OF A DEVICE (*Continued*)

Feature	Description
Battery power	You learnt about the importance of energy consumption earlier in the topic. A user will normally prefer to charge a device as infrequently as possible. Therefore, the battery power of a device is important. If a laptop has a battery life of two hours, this could be inconvenient for a user who needs to use their laptop for several hours in a day when they don't have the ability to charge it. So why not just build long-lasting batteries into devices? The type of battery needed by most devices can be quite large. To make a battery longer-lasting normally means it needs to be made larger. This can affect the portability of a device. Most modern batteries use a liquid electrolyte (lithium). Battery manufacturers are developing solid-state batteries, which use solid electrolytes, are more efficient and, so, can be made lighter and more compact. Battery developments will affect the design and portability of future devices.
Miniaturisation	Miniaturisation is when more **transistors** are fitted onto an **integrated circuit (IC)**. This is so that a device can be made smaller but still retain a similar performance. ICs are used in devices called **embedded systems**. An example of an embedded system is a smart watch. It has a limited number of functions. However, it needs to be reasonably powerful to have a good level of performance for the user. An IC allows the device to remain small but gives it the ability to perform well. An example of miniaturisation is the Raspberry Pi, which is a single-board computer – i.e. it can perform all the functions of a computer, but on a single (miniaturised) board. Single-board computers will not have the same level of performance as a full-sized computer, but they can provide the same functionality.
Processor	The processor in a device is the piece of hardware that processes all the data and instructions that are input into the device. It has a lot of work to do and the performance of a device is heavily affected by the processor. If a device can process data and instructions quicker, it will have a higher level of performance. However, as with most hardware, if it needs to have a higher level of performance, this means it often needs to be larger. This will affect the portability of a device. There are two main aspects of a process that can affect its performance: its clock speed and the number of cores it has. The clock speed is measured in gigahertz (GHz). If a processor has a clock speed of 1 GHz it can process 1 billion instructions per second. Therefore, a computer that has a clock speed of 3.5 GHz can process more instructions per second than one with a clock speed of 2.4 GHz. Each core in a computer is responsible for processing a single instruction. A processor that has one core will be able to process one instruction at a time. A processor that has two cores (a dual core processor) can process two instructions at the same time. Therefore, the more cores a processor has, the more instructions it can process at the same time.
RFID	You completed an activity earlier in the topic that included researching the use of RFID as a security feature. RFID tags can be included in devices to keep track of them. They are helpful in making sure that devices or other items do not get lost. RFID tags are even included in things such as golf balls! This can help golfers find their golf balls more easily while on the golf course.

(*Continued*)

KEY TERMS

embedded system: a type of system that has a dedicated purpose

integrated circuit (IC): a set of small electronics circuits that are built onto a single electronic chip

transistor: a very small component in a computer that controls the flow of electricity

TABLE 1.1: FEATURES AND THEIR EFFECT ON THE DESIGN AND USE OF A DEVICE (*Continued*)

Feature	Description
NFC	NFC can be a convenient way of sharing data between devices that are in very close proximity of each other. People can use NFC between their mobile phones to share data such as photos. NFC is also the technology that is used in making contactless payments with devices like a phone.
Quick response (QR) code	A QR code is a type of barcode that contains data such as a link to a web page or online document. QR codes can be scanned using a digital camera on a device. The data contained in the QR code can then be processed by the device to allow the web page to be opened on the device. It could be beneficial to include a QR code on a product as a consumer can scan it to access a lot of information that would be very difficult to include on the product itself.
Connectivity	You learnt about different methods of connectivity earlier in the topic. The design of a device can be affected by the method of connection chosen. If a port or other hardware (e.g. a WNIC) needs to be built into a device, the device may need to be larger. If a device is designed to need a wired connection, this device will not be portable. Therefore, any devices that need to be portable must have a wireless connection.

KEY TERM

quick response (QR) code: a type of barcode that can store data such as a link to a website

TECHNOLOGICAL CONVERGENCE

You have learnt about many different technologies in this topic. These different technologies are often linked together within devices and between devices. The linking together of all these different devices, like mobile phones, desktop PCs and a range of smart devices, is often done using the internet. This huge network of devices is called the **internet of things (IoT)**. Any device that can be connected to the internet is part of the IoT. We can often go through our day very unaware about how many devices are part of our daily life and are connected in the IoT. Objects like the fridge, washing machine, toaster and even heating systems can all be part of the IoT. It is amazing to think how many of these devices are seamlessly connected to make our life more convenient. The IoT is a result of **technological convergence**. Technological convergence is when two or more devices are combined to create a single device, product or service.

KEY TERMS

internet of things (IoT): the everyday home devices (e.g. a fridge, washing machine, toaster etc.) and sensors (e.g. car park space monitors, temperature sensors) that are connected to the internet

technological convergence: when two or more devices are combined to create a single device, product or service

▶ A tablet is an example of technological convergence

One example of technological convergence is a smart TV. There are several different technologies that have been built into a TV to give it extra functions, for example the ability to surf the web and stream TV programmes and movies. Some smart TVs even have a built-in webcam; so, they can be used to make video calls.

SKILLS

CREATIVITY, ANALYSIS

ACTIVITY 8

The editor of an online technology magazine wants to include an article about technological convergence on their website. The editor has asked you to write this article. They need you to include a description of what is meant by technological convergence and describe three different examples of technological convergence. In your examples you should explain what kind of devices have been combined.

Make sure you have your article proofread by a friend. Ask your friend to give you feedback based on the following.

- Have you included enough information to fully explain technological convergence?
- What is the most interesting example that you have included in your article?
- What is one thing that they think you could improve about your article?

EMBEDDED SYSTEMS

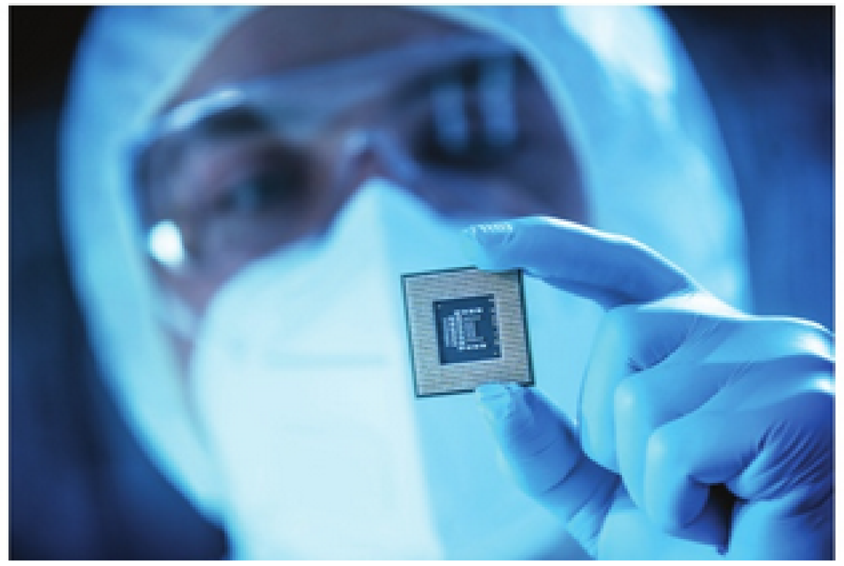
You learnt a small amount about embedded systems earlier in the topic when learning about miniaturisation. Embedded systems are used in many different devices. They are an example of technological convergence as they allow small systems to be embedded into everyday devices, often turning them into a smart device. There are several features that define an embedded system. An embedded system:

- has a single function or limited functions
- has dedicated hardware
- is not easily reprogrammed by the user
- is built into a single electronic chip.

KEY TERM

microprocessor: a small version of a processor where all the data is processed using a single or limited number of integrated circuits

The hardware that is used in an embedded system is called a **microprocessor**. It is simply a small version of a processor. It can do the same task as a processor, which is to process data and instructions that are input to the device. It is just a smaller version that does not have a high level of performance.



One example of an embedded system is a digital alarm clock. It has limited functions, such as displaying the time, allowing an alarm to be set and sounding an alarm at a set time. It can do this as it contains a microprocessor to enable these limited functions.

In addition to a digital alarm clock, some other examples of embedded systems include:

- robot vacuum cleaner
- smart TV
- smart watch

- security system
- air conditioning system
- electronic calculator.

Owing to the development of microprocessors, embedded systems can be created. Embedded systems can be used to add extra functions to items to make them more helpful and convenient in everyday life.

KEY TERMS

basic input/output system (BIOS): a set of instructions that are part of the firmware that control the input and output operations

bootloader: a program that is part of the firmware; it is responsible for loading the operating system when a computer is turned on

firmware: the basic instructions that allow a system to function

general-purpose computer: a computer that is used for a large range of different tasks

FIRMWARE

All embedded systems contain an important type of software called **firmware**. Firmware contains the basic instructions that allow the system to function, including those that help start up the system and for basic input/output tasks. The firmware in an embedded system is stored within the single chip it contains. This is different to a **general-purpose computer** where the firmware is stored in ROM. The firmware contains:

- the **basic input/output system (BIOS)**
- the **bootloader**.

The BIOS is a set of instructions used to boot up the computer. It initialises the hardware and gets it ready for the operating system to use. The bootloader is then responsible for loading the operating system.

FACTORS THAT ARE USED TO ASSESS THE PERFORMANCE OF A DEVICE

You have learnt about lots of different types of devices in this topic, and some of the features and functions of these devices. Performance is one of the key features of a device, but how can you assess the performance of a device? Table 1.2 shows the five factors that can be used to assess the performance of a device.

TABLE 1.2: FACTORS THAT ARE USED TO ASSESS THE PERFORMANCE OF A DEVICE

Factor	Description
Speed	The speed of a device refers to how quickly it can process instructions. The faster a device processes instructions, the faster it completes tasks like opening web pages, playing music and streaming a movie or video. Tasks like these can mean a large number of instructions need to be processed very quickly. The speed of the processor shows how many instructions it processes in a single second. If a processor has a speed of 2.4 GHz it can process 2.4 billion instructions a second. If a processor has a speed of 3.2 GHz it can process 3.2 billion. If one phone has a processor speed of 2.4 GHz and another phone has a processor speed of 3.2 GHz, it is likely that the one with the greater speed will have a higher level of performance. If a processor has more cores it may also be able to complete tasks more quickly. Each core processes a single instruction at a time; so, a processor that has multiple cores can process multiple instructions simultaneously. This means that if a task requires many instructions to be completed, and some of these instructions can be completed at the same time, the task will be completed quicker. This will improve the performance of the device.

(Continued)

TABLE 1.2: FACTORS THAT ARE USED TO ASSESS THE PERFORMANCE OF A DEVICE (*Continued*)

Factor	Description
Capacity	The capacity of a device refers to the amount of storage capacity it has. This means how much data it can store. A device that is able to store more data is often seen as a device that has a higher level of performance. This can be because it provides more convenience to the user.
Portability	You learnt about the portability of a device at the start of this topic. If it is important for a device to be portable, then a device that is small and lightweight will have a higher level of performance. This is because it is easier for the user to carry around. It is not necessary for all devices to be portable. For example, whilst there is little need for a smart TV to be portable, it is essential for a laptop.
Bandwidth	Bandwidth refers to the speed at which a network can transmit data. It is measured by the potential amount of data that can be sent across the network in a single second. For example, a network that has a bandwidth of 40 Gigabytes per second (Gbps) can send up to 40 gigabyte (GB) of data in a single second. If a network has a higher bandwidth, data can be sent between devices at a much quicker rate. This is because more data can be sent in a single second. This means the network has a higher level of performance, and this can allow the devices connected to the network to have a higher level of performance. This is because they can receive the data that they need more quickly.
Power efficiency	You learnt about the energy consumption of devices earlier in this topic. A device that uses energy more slowly to power it is often seen as a device with a higher level of performance. This is because it is more convenient for the user to not need to regularly charge the device. It is also more cost-effective for the user as it will use less electricity and, so, will also be kinder to the planet.

KEY TERM

bandwidth: the maximum amount of data that can be transferred from one device to another in a set amount of time

SKILLS

ANALYSIS, COLLABORATION

ACTIVITY 9

Find a partner to work with. Choose three different devices to assess in terms of their performance. These can be devices that you or your partner own. You could also choose devices that you would like to own, or that you think are important.

Assess each of your chosen devices by discussing the factors shown in Table 1.2. Write a report with your partner about your three chosen devices and how you have judged their level of performance.

EXAM TIP

Try to get as much experience as you can with a range of different devices before the exam. This will allow you to develop real-life experiences with devices, which will help you discuss them better in an exam question.

You have learnt about a wide range of important features and factors of devices in this topic. You have also learnt how to assess their performance. This knowledge can be used to help you select the correct device to meet the needs of an individual or organisation.

SKILLS

ANALYSIS, CRITICAL THINKING

ACTIVITY 10

Arun is an ICT student who needs a new device for his university projects. He will need to:

- record and edit video for his multimedia class
- build websites for his web design class
- carry his device from class to class and also to the library to study
- connect his device to other devices to share his projects
- store a lot of data on his device.

Arun is thinking of buying a mobile phone, a tablet computer or a laptop computer.

Analyse the suitability of each of these devices and write a recommendation for Arun, stating which device he should choose and why.

Priti is the owner of a restaurant. She wants her staff to have devices that they can use to take customers' orders at the table. The devices must:

- be available at each table in the restaurant
- have an excellent battery life
- be able to connect to a network so that the customers' orders can be sent to the restaurant kitchen.

Priti is thinking of buying each member of staff a mobile phone or a tablet. She is also thinking of installing a small touchscreen computer at the end of each table.

Analyse the suitability of each of these devices and write a recommendation for Priti, stating which device she should choose and why.

KEY TERMS

analogue: data that is continuous in nature and is the data that is processed by humans

binary: a number system that uses values 0 and 1

denary: a number system that uses values 0 to 9

digital: data that is discrete in nature and is the data that is processed by computers

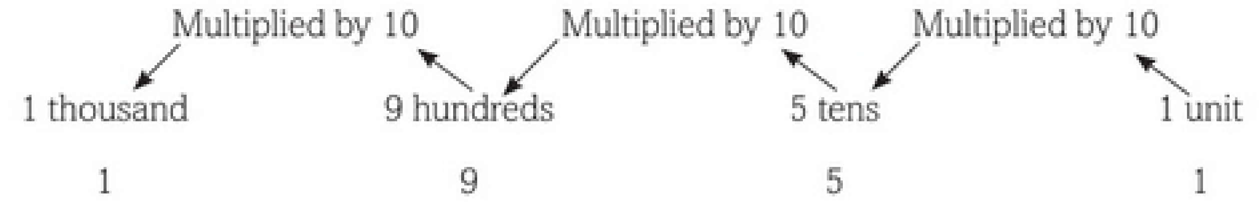
PROCESSING DATA

Devices need to process lots of different types of data such as text, numbers, sounds and images. All data that is processed by a device must be converted to **binary**. This is the only type of data that a device is able to process. Humans process data in **analogue** form, but a device processes data in **digital** form. Digital data is binary. It only has two values, which are 0 and 1.

Converting denary numbers to binary

You need to understand how data is converted to binary, and a simple way to do this is by looking at how numbers can be converted to binary. As humans, we mostly use a number system called **denary**. This number system has ten different values, which are the numbers 0 to 9. Denary numbers have to be converted to binary numbers to be processed by a device. How is this done?

Our denary number system has units that increase in multiples of 10. We can see this if we look at the number 1951. If we begin at the right-hand side of the number, we can see that each unit to the left increases by a multiplication of 10:



The binary number system has units that increase in multiples of 2. This means that the units start at 1 and increase in multiplications of 2. For example:

Multiplied by 2
Multiplied by 2
Multiplied by 2
8
4
2
1

Now you know what the units are for the binary number system, it is possible to convert a denary number into a binary number. Let's start with a simple example, the denary number 13. If a unit is needed in the binary number system it is marked with a 1; if it is not needed it is marked with a 0. We can start at the highest unit to see which units are needed. To convert the denary number 13, we first ask: is 13 greater than or equal to 8? The answer is yes, so we mark the unit 8 with a 1 and subtract 8 from 13:

8	4	2	1
1			

13 − 8 = 5

Now, we move to the next unit, which is 4. We now ask: is 5 greater than or equal to 4? The answer is yes, so we mark the unit 4 with a 1 and subtract 4 from 5:

8	4	2	1
1	1		

5 − 4 = 1

Now, we move to the next unit, which is 2. We ask: is 1 greater than or equal to 2? The answer is no, so we mark the unit with a 0.

8	4	2	1
1	1	0	

Now, we move to the next unit, which is 1. We ask: is 1 greater than or equal to 1? The answer is yes, so we mark the unit with a 1 and subtract 1 from 1.

8	4	2	1
1	1	0	1

1 − 1 = 0

We now know from this process that when the denary number 13 is converted to binary it is 1101.

Let's look at a second example. In the last example we looked at a 4-bit binary number; this time we will look at an 8-bit binary number. The units for an 8-bit binary number are:

128	64	32	16	8	4	2	1

We are going to convert the denary number 150 to binary. We start with the first unit, which is 128. We ask: is 150 greater than or equal to 128? The answer is yes, so we mark the unit with a 1 and subtract 128 from 150:

128	64	32	16	8	4	2	1
1							

150 − 128 = 22

We then move to the next unit, which is 64. We ask: is 22 greater than or equal to 64? The answer is no, so we mark the unit with a 0.

128	64	32	16	8	4	2	1
1	0						

We then move to the next unit, which is 32. We ask: is 22 greater than or equal to 32? The answer is no, so we mark the unit with a 0.

128	64	32	16	8	4	2	1
1	0	0					

We then move to the next unit, which is 16. We ask: is 22 greater than or equal to 16? The answer is yes, so we mark the unit with a 1 and subtract 16 from 22:

128	64	32	16	8	4	2	1
1	0	0	1				

22 − 16 = 6

We then move to the next unit, which is 8. We ask: is 6 greater than or equal to 8? The answer is no, so we mark the unit with a 0.

128	64	32	16	8	4	2	1
1	0	0	1	0			

We then move to the next unit, which is 4. We ask: is 6 greater than or equal to 4? The answer is yes, so we mark the unit with a 1 and subtract 4 from 6:

128	64	32	16	8	4	2	1
1	0	0	1	0	1		

6 − 4 = 2

We then move to the next unit, which is 2. We ask: is 2 greater than or equal to 2? The answer is yes, so we mark the unit with a 1 and subtract 2 from 2:

128	64	32	16	8	4	2	1
1	0	0	1	0	1	1	

2 − 2 = 0

We have no numbers left now, so we know the last unit must not be needed. We can mark it with a 0.

128	64	32	16	8	4	2	1
1	0	0	1	0	1	1	0

We now know from this process that when the denary number 150 is converted to binary it is 10010110.

EXAM TIP

You can do a quick check in an exam that can give you some indication that the binary number is the correct conversion for the denary number. If the denary number is an odd number then the binary number should have a 1 at the rightmost digit.

Converting binary numbers to denary

You have learnt how to convert denary numbers to binary. Binary numbers also need to be converted back to denary numbers by a device so that they can be displayed to the user. You are now going to learn how to convert binary numbers back to denary numbers.

Let's look at the binary number 01011101. We can see which units are needed for this binary number:

128	64	32	16	8	4	2	1
0	1	0	1	1	1	0	1

All we need to do to convert this binary number to denary is to add together all the values that are marked with a 1:

64 + 16 + 8 + 4 + 1 = 93

We can see that the binary number 01011101 converted to denary is 93.

SKILLS
PROBLEM SOLVING

ACTIVITY 11

Convert the following numbers from denary to binary:

- 7
- 10
- 51
- 99
- 180
- 256

Convert the following numbers from binary to denary:

- 0010
- 1001
- 1111
- 00111110
- 10001000

DATA FILE SIZE

Calculating the size of a file

The size of a file will depend on the type of data stored in the file. For example, a simple text document will have a very small file size, whereas a video will have a much larger file size.

The size of a file can be measured in several different units. The smallest unit that can be used to measure the size of a file is a byte. The smallest unit that can be used to measure the size of data is a bit. Table 1.3 shows how many units of data are equal to each unit of measurement for a file.

TABLE 1.3: UNITS OF MEASUREMENT FOR A FILE

Unit of measurement	Equal to
Bit	This is the smallest measure of data
Nibble	4 bits
Byte	8 bits
Kibibyte (KiB)	1024 bytes
Mebibyte (MiB)	1024 KiB
Gibibyte (GiB)	1024 MiB
Tebibyte (TiB)	1024 GiB

Examples

If the size of a file is 10 KiB, what would be the size of that file in bytes? You know from the table that 1 KiB is equal to 1024 bytes. Therefore, 10 KiB must be 1024 × 10 = 10 240 bytes.

If the size of a file is 2048 GiB, what would be the size of that file in tebibytes? 1024 GiB is equal to 1 TiB. Therefore, 2048 ÷ 1024 = 2 TiB.

These units of measurement for a file are the International Electrotechnical Commission (IEC) units. There is a second set of units that can be used to measure data and they are the ones you may be more familiar with for data storage. These are the international system of units (SI). They are:

bit
nibble
byte
kilobyte (kB)
megabyte (MB)
gigabyte (GB)
terabyte (TB).

In these units 1 kB is equal to 1000 bytes. 1 MB is equal to 1000 kB. The remaining units are all based on 1000 instead of 1024.

To calculate the size of a file we can use a text file as an example. Each character in a test file needs 1 byte of data (using American Standard Code for Information Interchange (ASCII)). If a text file has 3072 characters, the size of the file can be measured as:

24 576 bits (3072 × 8)
3072 bytes
3 KiB (3072 ÷ 1024) or 3 kB.

Calculating the time needed to transfer a file

You do not need to be able to calculate the exact time needed to transfer a file, but you do need to understand how to write an expression for the calculation. An expression shows how the time will be calculated without showing the final value.

Example

A student needs to send an image file to their friend's computer. They have connected their computer using an ethernet cable that has a 10 gigabit per second connection. The image file has a file size of 10 MiB.

The first part of the expression shows how the number of bits is calculated to show how many bits need to be sent across the connection. This would be 10 multiplied by 8 for bytes, 1024 for KiB and 1024 for MiB:

10 × 8 × 1024 × 1024

This is then divided by the second part of the expression, which shows a breakdown of how many bits are needed for the ethernet connection. This would be 10 multiplied by 1000 for kilobits, 1000 for megabits then 1000 for gigabits. No multiplication by 8 as it is a gigabit connection not a gigabyte one. The full expression would be:

$$\frac{10 \times 8 \times 1024 \times 1024}{10 \times 1000 \times 1000 \times 1000}$$

This expression shows, in seconds, how long it will take to transfer the file.

ACTIVITY 12

An employee has a file that is 100 MiB. How would you calculate how many bytes are in the file?

If the employee sends the file to another employee across a 5 gigabit connection, how would you calculate how long this would take in seconds?

SOFTWARE

Software is all the instructions that tell a computer what to do. There are several different types of software that you are going to learn about.

SYSTEMS SOFTWARE VS APPLICATION SOFTWARE

The two main types of software used by a computer are **systems software** and **application software**.

The purpose of systems software is to allow the computer to run efficiently. It helps to maintain the computer and perform tasks that the computer needs. It also allows the user to interact with the computer and control the hardware. Examples of systems software include:

- operating system
- antivirus
- firewall
- drivers
- disk defragmentation
- compression software.

KEY TERMS

application software: a type of software that provides services needed by the user

database: a collection of data that is related

software: the instructions that tell a computer what to do

systems software: a type of software that provides services needed by the computer

▼ Microsoft applications

Microsoft Apps
Excel
PowerPoint
OneNote
Skype
Word
OneDrive

The purpose of application software is to perform services for the user. It allows the user to use the computer to carry out tasks. Examples of application software include:

- word processor – to create a CV for a job, to produce business reports and so on
- spreadsheet – to create a budget for household bills, to track company expenditure, etc.
- **database** – to store a list of products and prices and to track sales
- photo editing software – to change the lighting effects on images
- video editing software – to create a video for an online advertising campaign
- web browser – to search for information on the internet
- games – to have some relaxation time.

KEY TERMS

driver: a type of systems software that allows devices to communicate with the operating system

interrupt: a type of signal that is sent to the processor in a computer to gets its attention

multitasking: when multiple tasks are executed by a computer at the same time

operating system: a type of systems software that manages the hardware and software in a computer

THE ROLE OF THE OPERATING SYSTEM

The **operating system** in a computer is a very important piece of software. It allows a user to interact with the computer as well as manages lots of processes within the computer. There are four main areas that the operating system manages; these are:

- managing devices
- managing processes
- managing users
- managing security.

Managing devices

The operating system manages devices using the peripheral devices that are connected to the computer. A computer cannot be used without input and output devices, so managing these devices is important. Some examples of peripheral devices managed by the operating system are:

- keyboard
- mouse
- monitor
- printer
- scanner.

Not all peripherals and computers are built to use the same computer-based languages. This means that there needs to be a system that can translate between the two to make them compatible. A **driver** is used for this purpose, and drivers are managed by the operating system. The operating system allows the correct drivers to be installed to make sure that the peripheral device and the computer can communicate effectively.

A peripheral device will send a signal called an **interrupt** to the computer when it needs to get the processor's attention about a task. The operating system manages these interrupt signals and decides which interrupts to send to the processor.

Managing processes

Thousands of processes are carried out each second in a computer. The operating system manages the process of moving data to and from storage and memory within the computer. It moves data in and out of the RAM. It makes sure that enough memory is allocated to each task that needs to be performed by the computer. It also makes sure that no two tasks try to access the memory locations in the RAM.

You will often want to do more than one task at a time on a computer. You could be listening to music while researching information on the web to do your homework. When multiple tasks are being done on the computer at the same time, the computer uses **multitasking**. This is when the processor is able to process more than one task at a time. The operating system decides which tasks need to be processed next and how long the computer should spend doing each task, before switching to another.

Managing users

There are several ways in which an operating system manages the needs of the user. These are as follows.

- It provides the user with an interface that they can use to interact with the computer. This is often a graphical user interface. This means that it has features like windows, icons, menus and pointers to help the user.
- It allows different users to create their own account on the same computer.

It allows users to use a range of actions with their files, such as open, close, move, delete and copy files.

It provides a platform to run application software that allows the user to use the computer for different tasks.

Managing security

The operating system can provide a range of security features for a computer. These include the following.

Allowing users to set a username and password for their account.

Allowing levels of access to be set for data that is normally linked to a username. This means that certain users can be restricted from seeing data on the computer or from doing anything more than viewing it. This can ensure that data is kept confidential if necessary. It can also make sure that data is not accidentally deleted by a user.

Running utility software such as antivirus, anti-malware and a firewall to keep a computer safe from various types of threats.

SKILLS
CREATIVITY

ACTIVITY 13

Create an infographic that could be displayed on your classroom wall to show the different roles of an operating system.

SOURCES OF SOFTWARE

Software is developed by thousands of individuals and organisations around the world. Table 1.4 shows the different sources from which software can be obtained.

KEY TERMS

free software: software that allows the user to study, modify and share the source code; a fee can be charged for the software

open-source software: software that allows the user to study, modify and share the source code; a fee is not charged for the software

proprietary software: software that is not distributed with the source code

TABLE 1.4: SOURCES OF SOFTWARE

Source	Description
Free	**Free software** is software that is developed and then released, allowing the user a wide range of freedom to study the source code for the software, modify it and share it. The reference to 'free' in the software title does not mean that the software is distributed without a fee. It refers to the freedom that the user has with the software. Software that is available without a fee is actually called freeware. One example of free software is Firefox.
Open-source	**Open-source software** is similar to free software. It allows the user to study, modify and share the software code. Open-source software is nearly always provided without a fee. Audacity is an example of open-source software. Open-source software relies on large communities to develop and test the software to improve it.
Proprietary	**Proprietary software** is software that is not distributed with the source code. The source code is protected, meaning that the user cannot see it and is not able to study, modify or redistribute the code. One benefit of proprietary software is that it is extensively tested by the developer, and the developer will also offer support for the software. One example of proprietary software is Microsoft Office.
Creative Commons	**Creative Commons** is an organisation that provides licences so that people can distribute their software. The licences use a standard way to give permission on how their software can be used by setting copyright terms. There are four different types of Creative Commons licence. • Attribution – to use the software code you must credit the author of the software. • Share Alike – to create a modification of the software, it must also have the share alike licence. • Non-commercial – the software must only be used for non-commercial purposes. • No derivatives – a user can modify the software but they cannot share a modification of the software.

(*Continued*)

TABLE 1.4: SOURCES OF SOFTWARE (*Continued*)

KEY TERM

Creative Commons: an organisation that provides different types of licences for software distribution

Software licences

Table 1.5 shows the different types of licences under which software is distributed.

TABLE 1.5: SOFTWARE LICENSING MODELS

Licence	Description
Single user	This type of licence allows a single user to use the software. It is typically designed so that the software can only be installed and used on a single computer.
Multiple user	This type of licence allows the software to be installed and used on a set number of computers. For example, a five user licence can be installed on five different computers. The set number of computers often needs to be within the same building.
Institutional	This type of licence allows all computers that belong to an organisation to have software installed on them. The fee for this type of licence is normally based on an approximate number of users in the organisation. For example, 1 to 20 users would mean a particular set fee, while 21 to 40 users would be a higher fee.
Fixed term	This type of licence allows software to be installed and used for a fixed period of time. After the fixed period of time the payment for the software can either be cancelled or it can be renewed for another fixed period of time. This type of licence has now been adopted by many companies, including Microsoft. Microsoft Office requires a subscription from users, normally on a yearly basis.
Indefinite	An indefinite software licence is also known as a perpetual licence. It allows a user to use the software indefinitely. With this type of software, the user normally pays a single fee upfront that allows them to buy the software, then the user can use the software on an ongoing basis without any further payment.
Network	This type of licence is based on where the software is installed. It means that the software is installed on a server in the network, and any client computers that are part of the network and connected to the server can use the software.

Updating software

Once software has been installed on a computer it is likely that it will need to be regularly updated. This is to make sure that the software has the most up-to-date code that is available. Developers release software updates for several reasons, including to:

- fix any bugs that have been reported
- add new features to the software
- improve the security of the software
- change the look of the interface for the software.

Table 1.6 shows the different ways that software updates can be released by a developer.

TABLE 1.6: TYPES OF SOFTWARE UPDATE

Type of update	Description
Patch	A **software patch** is normally a quick-fix that is released by a software developer to fix a set of errors or improve functionality. It is often used to quickly fix a security vulnerability that has been found in the software. Open-source software can often be poorly supported with software patches if the community stops developing the software and updating it.
Automatic	An **automatic software update** is an update that does not require the user to continually consent to the software being updated. The user can set the software to have automatic updates. This means that the software will be updated automatically every time the software developer releases an update. Large software developers like Microsoft are constantly updating their software and will recommend that automatic updates are set. This is to make sure that a user's software is without errors and known security vulnerabilities are fixed.
Upgrade	A **software upgrade** is normally a whole new version of an existing software that contains lots of new features and functionality. This type of update normally completely replaces the old version of the software with a new version.

KEY TERMS

automatic software update: a setting that can be applied to software updates to install them without a user's intervention

compatibility issues: problems that can arise when an update to software causes the software to not function properly

software patch: a set of changes used to fix errors or improve functionality in software

software upgrade: a whole new version of existing software that contains new features and functionality

One issue that can arise from updating software is **compatibility issues**. This means that the update to the software causes the software to not function properly. One example of this is the software update being designed to run on newer hardware, so it doesn't run efficiently on older hardware. Another example of this would be an operating system being updated and no longer supporting older hardware or applications.

SKILLS
ANALYSIS, REASONING

ACTIVITY 14

You need to have the skills to select software to meet the needs and requirements of an individual or organisation. This activity will help you.

Selma is a software developer who has a new application that she wants to release. She needs to choose which type of software she wants to release. She could release the application as free software, open-source software or proprietary software, or choose a Creative Commons licence. Selma is worried about her source code being stolen, but likes the idea of people being able to modify her software to suit their needs.

Analyse the suitability of each of these software types and write a recommendation for Selma to state which one she should choose and why.

THINKING LIKE AN IT SPECIALIST

Software development is a huge industry that creates thousands of new applications each year. Many of these applications are for mobile devices. Some of these applications will be free software, some will be open source, some will be proprietary. Can you think of an idea for an application? Would you release your application as free software, open-source software or proprietary software? Why?

CHECKPOINT

1. What are two features of a device that make it more portable?
2. What are two ways in which devices can be connected?
3. What is a biometric device and what kind of feature is it an example of?
4. What is a microprocessor and what is its role in an embedded system?
5. How can RAID storage be used to increase the performance of a computer?
6. What are two examples of systems software?
7. What are two examples of application software?
8. What is the Creative Commons?
9. What is an institutional software licence?
10. What is a software patch?

SKILLS
ANALYSIS, INTERPRETATION

EXAM PRACTICE

1. A laptop uses firmware.

(a) State the purpose of firmware. [1 mark]

(b) Give one example of an item that is present in firmware. [1 mark]

(c) A printer is connected to the laptop.

Identify the most appropriate type of wired connection to connect the printer to the laptop. Write the correct letter in your exercise book. [1 mark]

A HDMI cable
B USB cable
C Ethernet cable
D VGA cable

2. A computer uses systems software and application software.

(a) Explain the difference between systems software and application software. [2 marks]

(b) The computer has an operating system. One role of the operating system is to manage users.

(i) Give two ways an operating system manages users. [2 marks]

(ii) Give one other role of the operating system. [1 mark]

3. A software development company releases their software as proprietary software.

(a) Explain what is meant by proprietary software. [2 marks]

(b) A finance organisation buys an institutional licence for the software.

Describe what the institutional licence allows the finance organisation to do. [2 marks]

4. An employee has a text file that has 12 288 characters. Each character in the file is 1 byte.

(a) Calculate the size of the text file in kibibytes. [1 mark]

(b) Calculate the size of the file in bits. [1 mark]

(c) The employee sends a 30 GiB file to another employee across their network that uses a 10 megabit per second ethernet connection.

Construct an expression to show how long, in seconds, it should take to transfer the file. [3 marks]

TOPIC 2
NETWORKS

Connection and communication are a big part of our lives and society. The development of networks has led to more opportunities for us to connect and communicate. In this topic you will learn about many different types of networks and the components that are used to create them. This includes the different kinds of connection that we can make between our devices, such as wired and wireless connections. The development of networks has also created security issues when using our devices. You will learn what some of these security issues are and how we can help keep our devices safe when using a network.

TOPIC 2 NETWORKS

LEARNING OBJECTIVES

By the end of this topic you should be able to understand:

- the role of a range of different network components
- how different network transmission media are used and how they function
- the features and functions of different network models
- a range of ways the performance of a network can be measured
- the purpose and features of different communication methods, like ethernet, Wi-Fi, Bluetooth, cellular, infrared and ZigBee®
- a range of network standards and protocols, including their purpose and how they function, and the role of an IP address and a MAC address in a network
- how to design a network to meet a set of requirements, including the location of devices
- why a network needs to be kept secure and how this can be done using hardware and software security solutions.

GETTING STARTED

Find a partner to work with; you are going to play a game. Write down these key terms on different pieces of paper:

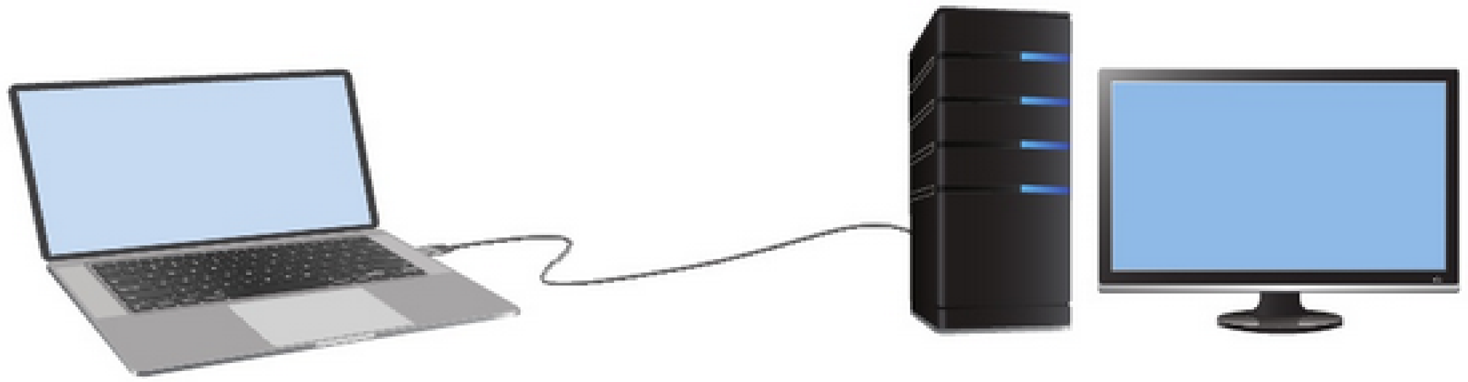
- network
- device
- local area network (LAN)
- internet
- Wi-Fi
- Bluetooth
- router.

Fold each piece of paper in half and place them into a container. Take it in turns to pull out a piece of paper and look at the key term. You are allowed one sentence to explain the key term to your partner, and they have two guesses to try to get it right.

At the end of the game, tell your partner which key term you think they described the best. Also tell them which key term you struggled with the most from their description.

WHAT IS A NETWORK?

A network is simply two or more devices that have been connected to each other.



▶ A simple network

NETWORK COMPONENTS

A network can contain a range of different network components, with each having a defined role in the network. Table 2.1 shows some of these components.

KEY TERMS

data collision: when two devices on a network send data simultaneously the data packets can collide

data packet: the unit of data that contains the information to be transmitted across a network

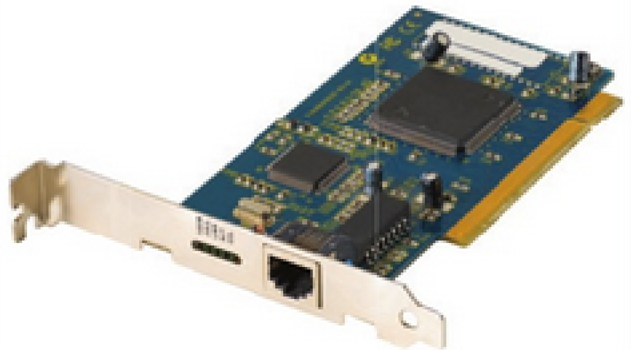
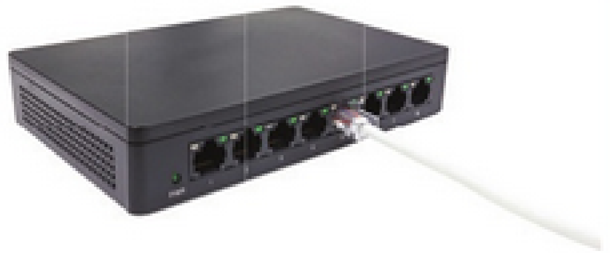
ethernet: a protocol used to connect devices in a LAN using a wired connection

hub: network hardware that devices connect to so that they can send data to each other in a network

local area network (LAN): a type of network that is normally located within a single building or site

network interface card (NIC): a piece of hardware that is built into a device; it allows it to have a wired or wireless connection

TABLE 2.1: NETWORK COMPONENTS

Network component	Description of role	What it looks like
Network interface card (NIC)	The main component that any device needs to connect to a network is a network interface card (NIC). This is a type of circuit board that is built into a device. A basic NIC allows a device to connect to a network using a wired connection. The wired connection is normally an **ethernet** connection. A wireless network interface card (WNIC) allows a device to connect to a network using a wireless connection.	
Hub	A hub is the most basic way of connecting several devices in a **local area network (LAN).** A hub receives **data packets** and forwards them on to all devices that are connected to the hub. Only the intended destination device will accept the data sent; the other devices connected to the hub will just ignore it. Each device is connected to a hub using a wired connection. The type of wired connection normally used for a hub is either an ethernet connection or a USB connection. There is a lot of unnecessary traffic in a network where hubs are used. This can cause a lot of **data collisions**, which can cause errors to occur and reduce the performance of the network.	

(*Continued*)

KEY TERMS

IP address: a unique address that is given to a device when it connects to a network

media access control (MAC) address: a unique address that is given to the NIC in each device at the point of manufacture

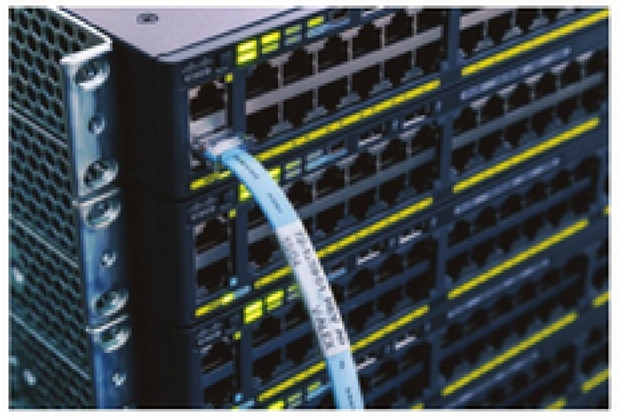
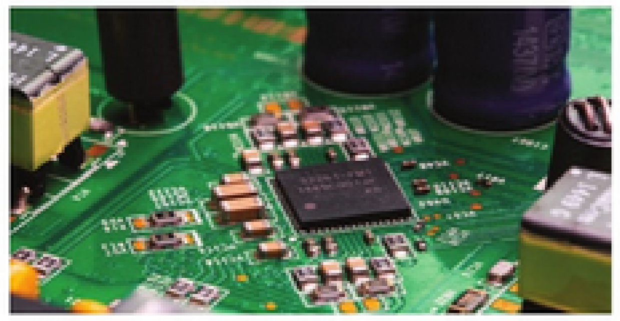
modem: network hardware that converts between analogue and digital signals

router: network hardware that reads IP addresses to send data to the correct network or device

routing table: a database that stores all the known routes across a network to different devices

switch: network hardware that reads MAC addresses to send data to the correct device

TABLE 2.1: NETWORK COMPONENTS (*Continued*)

Network component	Description of role	What it looks like
Switch	A switch was developed as an improvement to the inefficiency of a hub. A switch is also used to connect several devices in a LAN network. The major difference between a switch and a hub is that instead of transmitting data packets to all devices connected to the switch, it only sends the data packets to the intended destination. It does this by using a device's **media access control (MAC) address**. A switch records all the MAC addresses of devices that are attached to it and stores them in a table. It looks at the data packet to find the destination MAC address and forwards the data packet on to this device only. This reduces the amount of traffic in the network and as a result reduces the number of data collisions. This improves the performance of the network.	
Router	A router is used to forward data packets between devices in a network. It uses a special database called a **routing table** to send data packets along the correct route to their intended destination. It does this by reading the **internet protocol (IP) address** of the destination device that is stored in the data packet. It then sends the data packet towards the destination address and selects the most suitable route for it to take.	
Modem	Modem is short for modulator-demodulator. It is used to convert analogue signals to digital and vice versa. For example, it takes analogue signals that are sent down telephone wires across the internet and converts them to digital signals to be processed by a device. A modem is built into all modern devices that can be connected to the internet.	

(*Continued*)

KEY TERMS

bridge: network hardware that is used to connect one LAN to another LAN

gateway: network hardware that acts as a translator between the two networks, converting data from one protocol to another to allow the networks to communicate with each other

protocol: a set of rules that govern the transfer of data

wide area network (WAN): a type of network that is normally spread across several buildings or sites; the largest example is the internet

TABLE 2.1: NETWORK COMPONENTS (*Continued*)

Network component	Description of role	What it looks like
Gateway	A gateway is used to connect one LAN network to another LAN network, or a LAN network to a **wide area network (WAN)**. One network will often use different **protocols** to another network. This means that when they are connected, they are not able to communicate effectively with each other. It's like they are speaking two different languages. A gateway acts as a translator between the two networks, converting data from one protocol to another to allow the networks to communicate with each other.	
Bridge	A bridge is used to connect one LAN to another LAN. It does this to create one larger LAN. An organisation such as a university may use a bridge. The university may have many LAN networks in different buildings and sites around the university grounds. The LANs may be too far apart to work effectively as one single large LAN, so a bridge can be used to join each of the LAN networks to create a large LAN. A bridge works in a very similar way to a switch. It reads the MAC address of a device from the data packet and forwards it on to its correct destination. If the bridge sees that the MAC address is for a device in another connected LAN network, it will forward the data packet across the bridge to that network.	

(*Continued*)

KEY TERMS

repeater: a network device that is used to boost the strength of an electrical signal

server: a central device in a network that stores data and can also provide services; it accepts and responds to requests that are made for different types of data

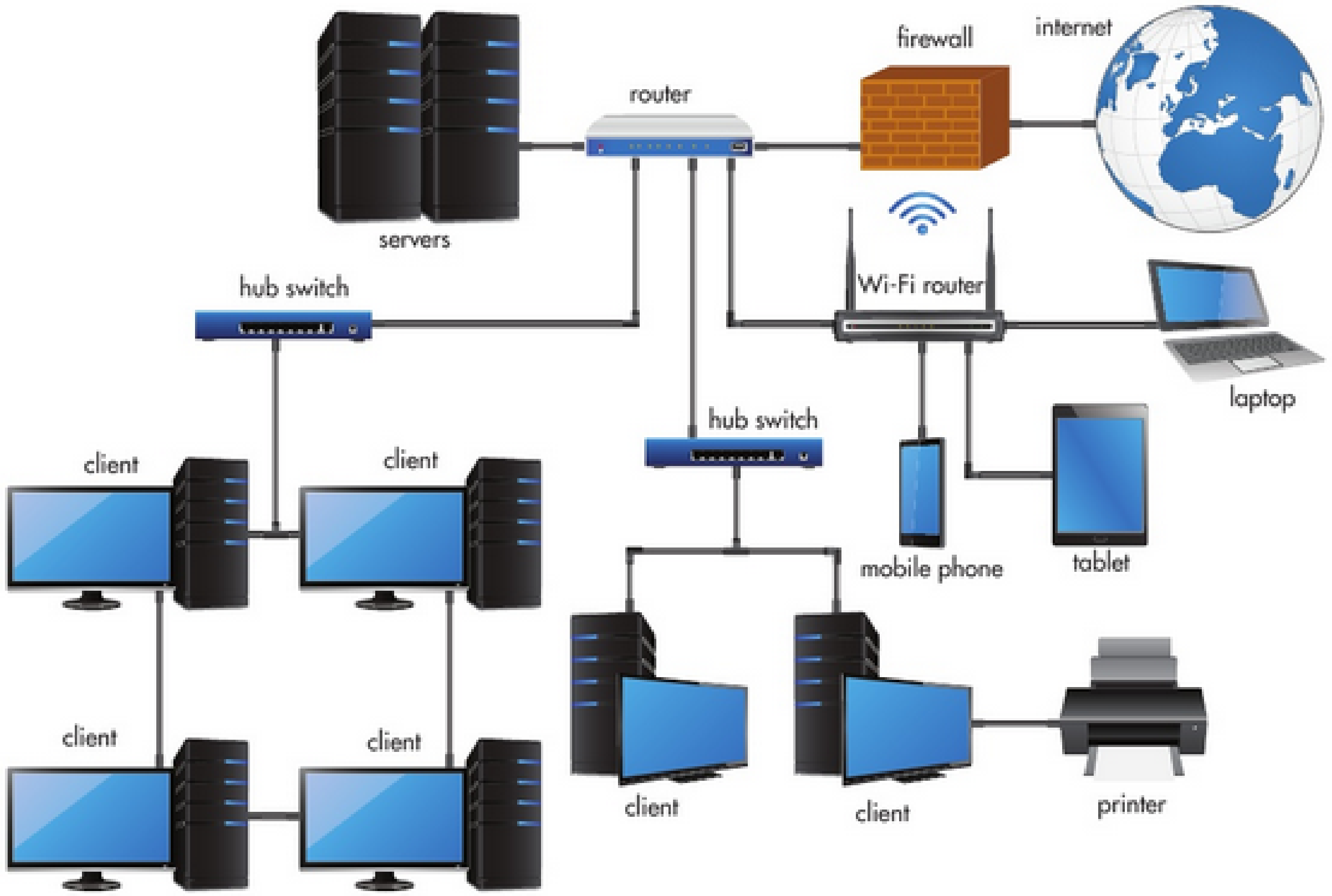
transmission media: the types of cables or signals that are used to transmit data across a network

wireless access point (WAP): network hardware that transmits and receives radio wave signals

TABLE 2.1: NETWORK COMPONENTS (*Continued*)

Network component	Description of role	What it looks like
Repeater	Electrical signals are transmitted around a network to send data. Shortly after an electrical signal is sent from a device in the network it will start to get weaker. This is because electrical signals are prone to interference caused by the type of **transmission media** used. You will learn more about the different types of transmission media later in this topic. If the signal gets too weak it will not be able to reach the intended destination device. A repeater is used to boost the strength of an electrical signal, meaning it can travel further across a network. A repeater can even reconstruct an electrical signal that has become distorted due to being too weak.	
Server	A server is a device that accepts and responds to requests that are made for different types of data that can be stored on the server, or for services provided by the server. A common example of a server is a web server. This type of server stores the web pages for a website and responds to requests for the web pages by sending the data for the web page to the requesting device. Servers are single board devices, like a computer that has a single board called the motherboard. Servers are often stacked together in a rack so that they can provide many different services like storing files, managing printing jobs and managing web page requests.	
Wireless access point (WAP)	An NIC allows a device to connect to other components in a network using an ethernet connection. However, if a device has a WNIC it will need to connect to the network using a wireless connection point. This wireless connection point is created using a wireless access point (WAP). A WAP receives wireless radio wave signals transmitted by devices with a WNIC and transmits them to their intended destination.	

It is possible for some network components to be combined into a single device or built into other devices. For example, if you think of the device that you call a router in your home or school, this is normally a combination of a modem, a router, a gateway and a wireless access point. Many also contain a switch as well. All routers have to be set up to act as a gateway, but note that not all gateways are routers!



▶ Figure 2.1 An example of a network (see page 42 for an explanation of 'client')

SKILLS

INTERPRETATION, CREATIVITY

ACTIVITY 1

Consider the network components that you have just learnt about. Choose three components that you are most interested in. Create a poster that could be displayed in your classroom to explain the role of these three components. Include a diagram on your poster that shows the role of each component.

NETWORK DESIGN AND IMPLEMENTATION

NETWORK TRANSMISSION MEDIA

Devices can be connected to a network using a wired connection or a wireless connection. The types of wired connection that can be used are:

- twisted pair cable
- coaxial cable
- fibre optic cable
- powerline.

The types of wireless connection that can be used are:

- microwave
- radio wave
- light
- satellite.

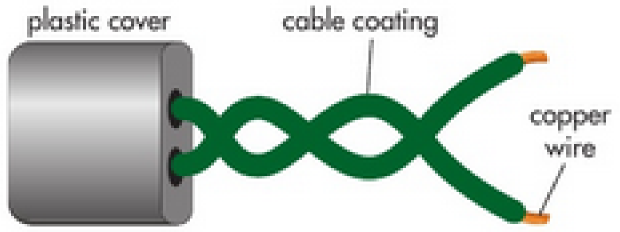
KEY TERMS

coaxial cable: a copper cable that has a single central wire with insulating layers; it can be used to transmit data across a network

twisted pair cable: a copper cable that has two twisted wires; it can be used to transmit data across a network

Twisted pair cable

A **twisted pair cable** is a type of copper cable that has two copper wires. Each copper wire has an insulated layer. The two copper wires are twisted around each other, and there is an outer plastic layer covering them.



▲ Figure 2.2 The layers of a twisted pair cable

A twisted pair cable can be unshielded or shielded. The diagram shows an unshielded twisted pair cable. A shielded twisted pair cable has a copper shield layer around the twisted wires, which is then covered by the plastic outer coating.

There are several characteristics of twisted pair cables.

They transmit data using electrical currents.

They can transmit data up to a speed of 10 Gbps.

They are cheap to manufacture.

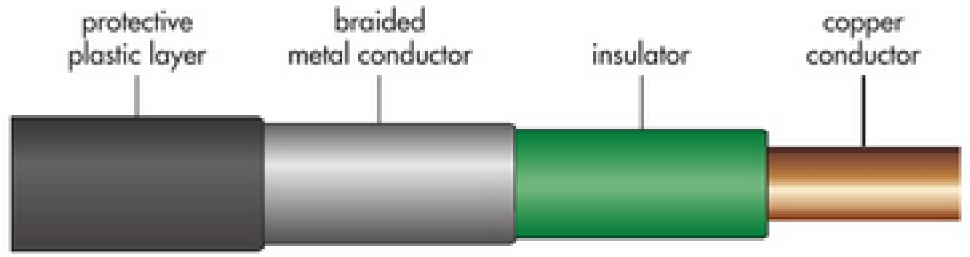
They can suffer from electromagnetic interference, so are only used to carry data over short distances.

They are easy to install.

They are mainly used for telephone communications and for ethernet connections in a local area network within one building.

Coaxial cable

A **coaxial cable** is a type of copper cable. It has a central copper wire that has an insulated layer, a metal conductor layer and a protective plastic outer layer.



▶ Figure 2.3 The layers of a coaxial cable

There are several characteristics of coaxial cables.

They transmit data using radio frequency signals.

They can transmit data at speeds of up to 10 Mbps.

They are cheap to manufacture, but not as cheap as twisted pair cables.

They are less prone to electromagnetic interference because of the braided metal conductor layer.

They are difficult to install, as they can be stiff and difficult to bend.

They are mainly used in cable TV connections and sometimes in internet connections.

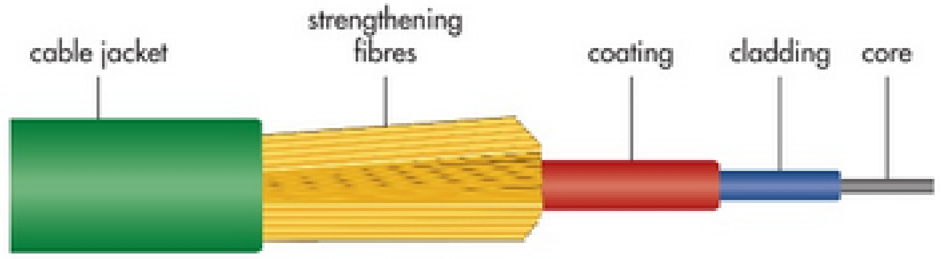
KEY TERMS

fibre optic cable: a cable that is made up of fine strands of glass or plastic

powerline: a device that plugs into a power socket and transmits data over the electrical wiring to another device on the same circuit

Fibre optic cable

A **fibre optic cable** has many very thin strands of glass or plastic that make up the core of the cable. The core is covered by a cladding, which is covered by a plastic coating. This plastic coating is covered by strengthening fibres and then an outer coating.



▲ Figure 2.4 The layers of a fibre optic cable

There are several characteristics of fibre optic cables.

They transmit data using light.

They can transmit data up to speeds of 1 Gbps.

They are expensive to manufacture.

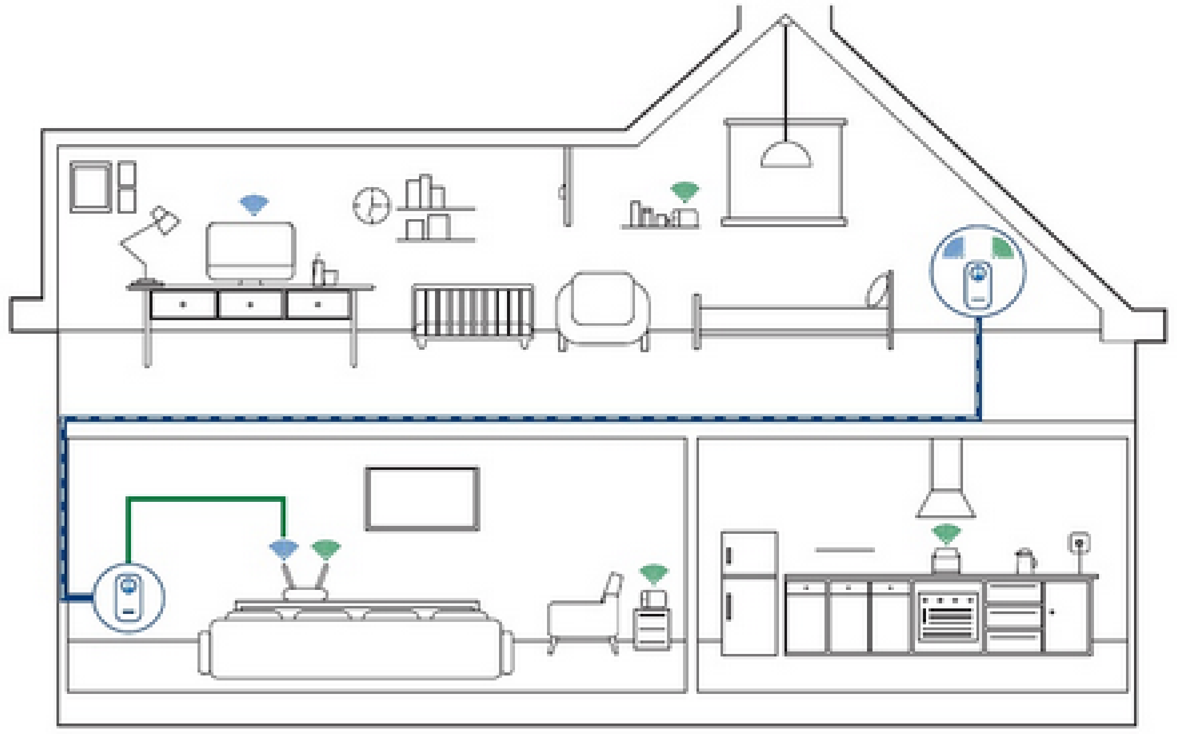
They are not prone to electromagnetic interference.

They are difficult to install, as the cables can be fragile.

They are often used for internet connection and transmitting data over long distances.

Powerline

Powerline is a type of connection that is created using adapters that are plugged into an electrical plug socket. When a powerline adapter is plugged into a socket, it transmits data over the electrical wiring in a building. Two powerline adapters are used to create a connection between the two points.



▶ Figure 2.5 A powerline connection

There are several characteristics of powerline connections.

They transmit data using electrical wiring.

They can transmit data at speeds of up to 2 Gbps.

They are relatively expensive to manufacture.

KEY TERMS

electromagnetic waves: a form of radiation that travels through the air

infrared light: electromagnetic radiation with wavelengths longer than visible light but shorter than radio waves; it can transmit signals only over very short distances

microwaves: electromagnetic radiation that can be used to create a wireless connection in a network; they operate only over short distances

radio waves: electromagnetic radiation that can be used to create a wireless connection in a network; they operate over longer distances than microwaves

They are prone to electromagnetic interference and are limited by the condition of the electrical wiring in the building.

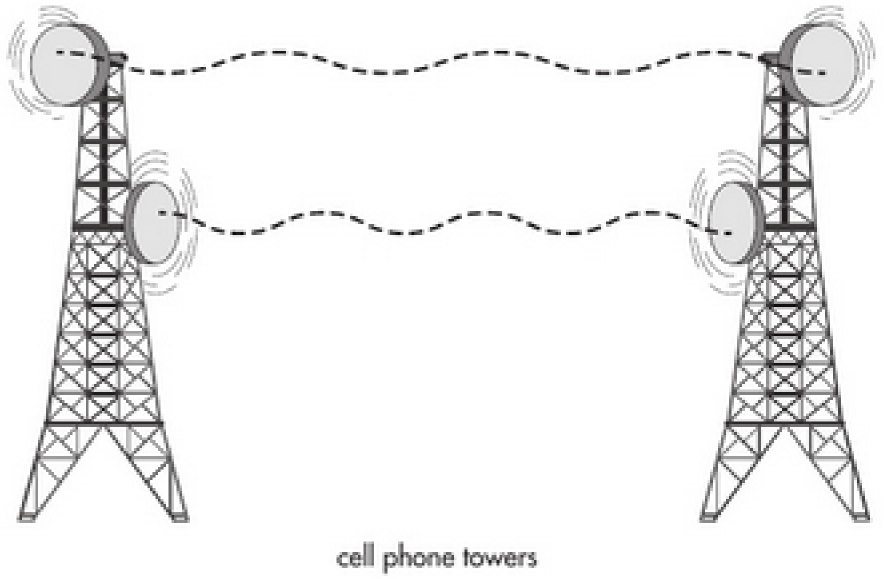
They are very easy to install, as they can be simply plugged into any electrical socket.

They are often used in large houses and buildings to allow data to travel more effectively over a longer distance. This can include extending an internet connection to an area that gets poor coverage in a house.

Microwaves

Microwaves are **electromagnetic waves** that can be used to create a wireless connection between technologies. Microwaves are point to point, which means that they can only travel in a straight line from one technology to another. Microwaves are high-frequency waves (between 1 GHz to 300 GHz) that are short in wavelength. For this reason they are normally used to transmit data over relatively short distances where there is a direct line of sight from one technology to another.

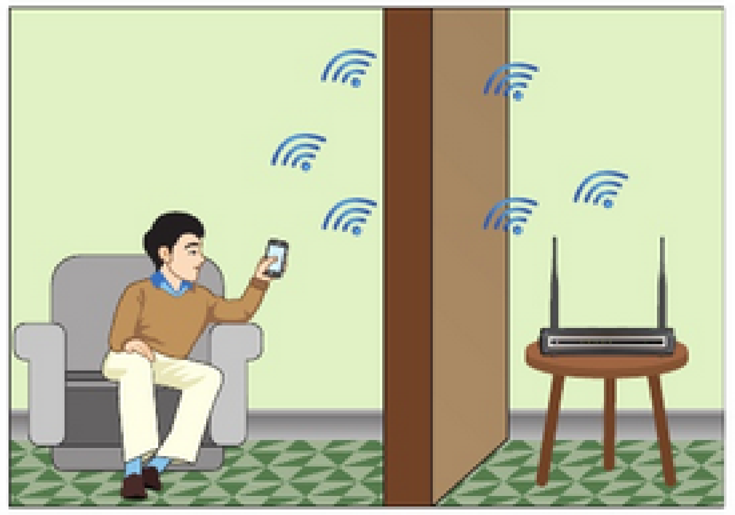
Communications between towers such as cell phone towers is normally done using microwaves. This is because microwaves are unidirectional, so they can be focused like a beam of light and are not scattered around. Neighbouring cell towers are positioned so that there is a direct line of sight between them.



▶ Figure 2.6 Microwaves can be used to transmit data from one cell phone tower to another

Radio waves

Radio waves are also electromagnetic waves that are used to create a wireless connection between technologies. Radio waves are omnidirectional, which means that they can travel in all directions and do not need a line of sight. Radio waves are lower in frequency than microwaves (3 kHz to 1 GHz) and are longer in wavelength. They are normally used to transmit data over both short and long distances, and they can pass through solid objects such as walls.

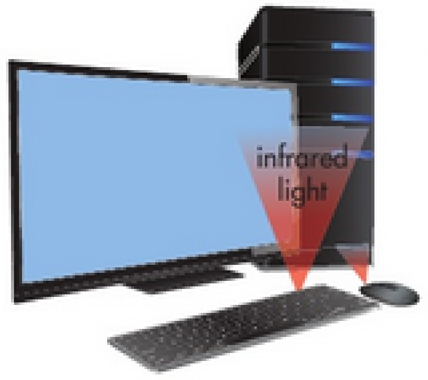


▲ Figure 2.7 Radio waves can be used to transmit data between a WAP and a mobile phone

Wireless LAN networks use radio waves to communicate. Radio waves can be sent from devices such as a WAP to other devices within range. These devices could be in a different room in a building, as radio waves can travel through walls.

Light

Infrared light can be used to transmit data wirelessly from one device to another. A beam of infrared light is directed through the air from one device to the other, so there needs to be a direct line of sight between the devices. The wavelength of infrared light is longer than the wavelength of visible light, but shorter than radio waves. Infrared is used to transmit data over short distances because it needs a direct line of sight and cannot travel through any obstacles.



▲ Figure 2.8 Infrared can be used to wirelessly connect a keyboard to a computer

KEY TERMS

ground station: a base on Earth that receives and transmits radio waves to a satellite

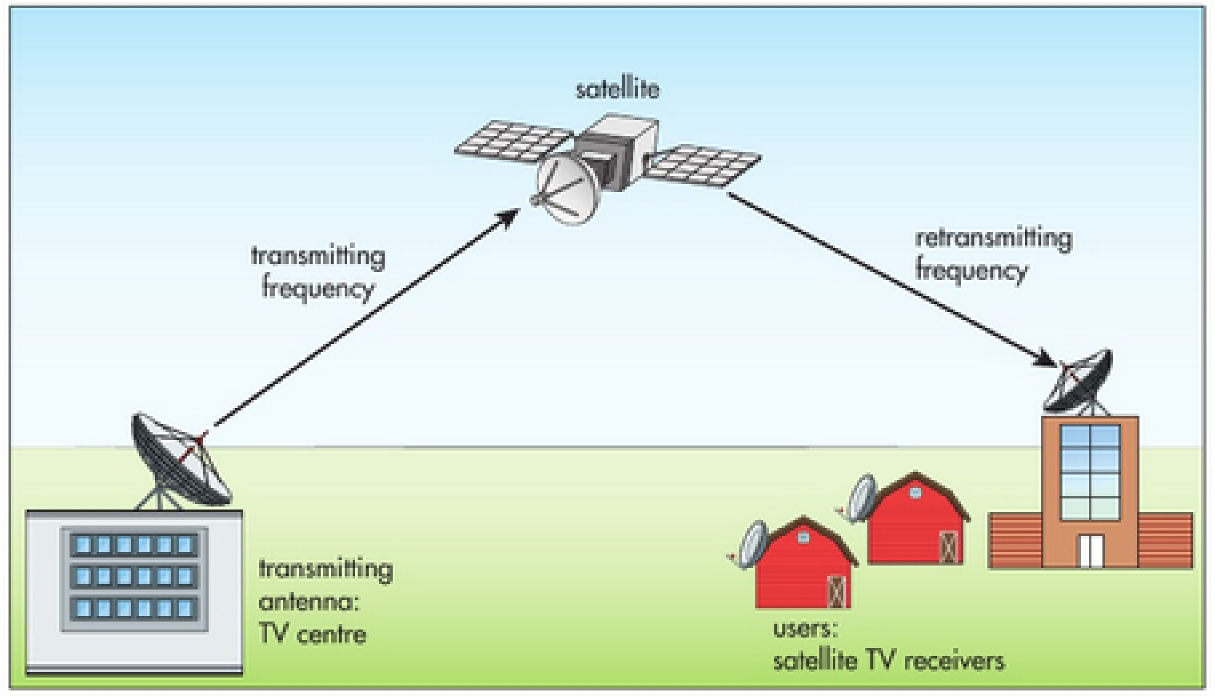
satellite: a device that orbits the Earth and uses radio waves to transmit data

Infrared is sometimes used as an alternative way to connect peripheral devices such as a keyboard and mouse to a computer. However, it is not a very popular method for connecting devices as an infrared signal is quite weak and requires a direct line of sight between the peripheral and the computer.

Satellite

A **satellite** is an artificial body that is sent into space to orbit the Earth. Satellites transmit data using radio waves. Radio waves are used to transmit data to the satellite from a base on Earth called a **ground station**. The radio waves can travel through each layer of Earth's atmosphere to the satellite. The satellite can also transmit radio waves to the ground station on Earth.

The most common use of satellite communications is in global positioning systems (GPSs). If a person has a portable device that has GPS, they will normally be in the line of sight of at least 10 satellites at any one time. A process called triangulation can be used to find a user. This is when at least three satellites are used to pinpoint the location of a user.


satellite
transmitting frequency
retransmitting frequency
transmitting antenna: TV centre
users: satellite TV receivers

▶ Figure 2.9 Radio waves can be transmitted between a ground station and a satellite

SKILLS

INTERPRETATION, CREATIVITY

ACTIVITY 2

Create a revision resource for the exam about the different types of transmission media that can be used in a network. You could create:

- flash cards
- an audio podcast
- a multiple choice quiz
- a mind map.

Swap your revision resource with a partner. Use your partner's revision resource to refresh your memory of the different types of transmission media.

Was there anything in their revision resource that you hadn't thought of?

Was there anything you had included that they hadn't?

TYPES OF NETWORK MODELS AND METRICS

You have learnt about the different ways that data can be transmitted in a network, but you also need to know the different kinds of structure a network can have. There are four main ways that a network can be structured:

- client–server network
- peer-to-peer network
- ad hoc network
- tethering.

KEY TERMS

client: a computer in a client–server network that is not a point of control in a network; it accesses data and services on a server

client–server network: a type of network that has central servers working as a point of control in the network and client computers that connect to them

CLIENT–SERVER NETWORK

There are two types of devices in a **client–server network**: the **client** computers and a server. A server is a computer that is a point of control in the network – it 'serves' the client computers that connect to it. The server will have a dedicated role in the network, for example:

- file server – stores and manages files
- web server – stores web pages and processes requests for the web pages
- print server – receives print jobs and transmits them to the correct printer.

Each client computer that is connected to the network sends requests to the servers in the network for data or services. Table 2.2 shows some benefits and drawbacks of a client–server network.

TABLE 2.2: BENEFITS AND DRAWBACKS OF A CLIENT–SERVER NETWORK

Benefits	Drawbacks
Security for the network can be carried out by a server. Each user does not need to be solely responsible for this.	A server is an expensive piece of technology to buy. A client–server network can need several servers, so it can be expensive to set up.
Updates for any software can be carried out by a server. Each user does not need to update their own computer. An organisation can be sure that everyone is using the same version of software. This can help avoid compatibility issues.	The network will need specialist technicians to maintain the servers and the network. This means they can be expensive to maintain.
A backup of all files can be created from the data stored on the server. Each user does not need to remember to back up their own files.	If a server fails then the users of the client computers in the network will not be able to access the resources and services provided by that server.
If a client computer crashes, it will not affect the rest of the network. The user will also be able to use another client computer in the network and access all the data and resources they were using.	

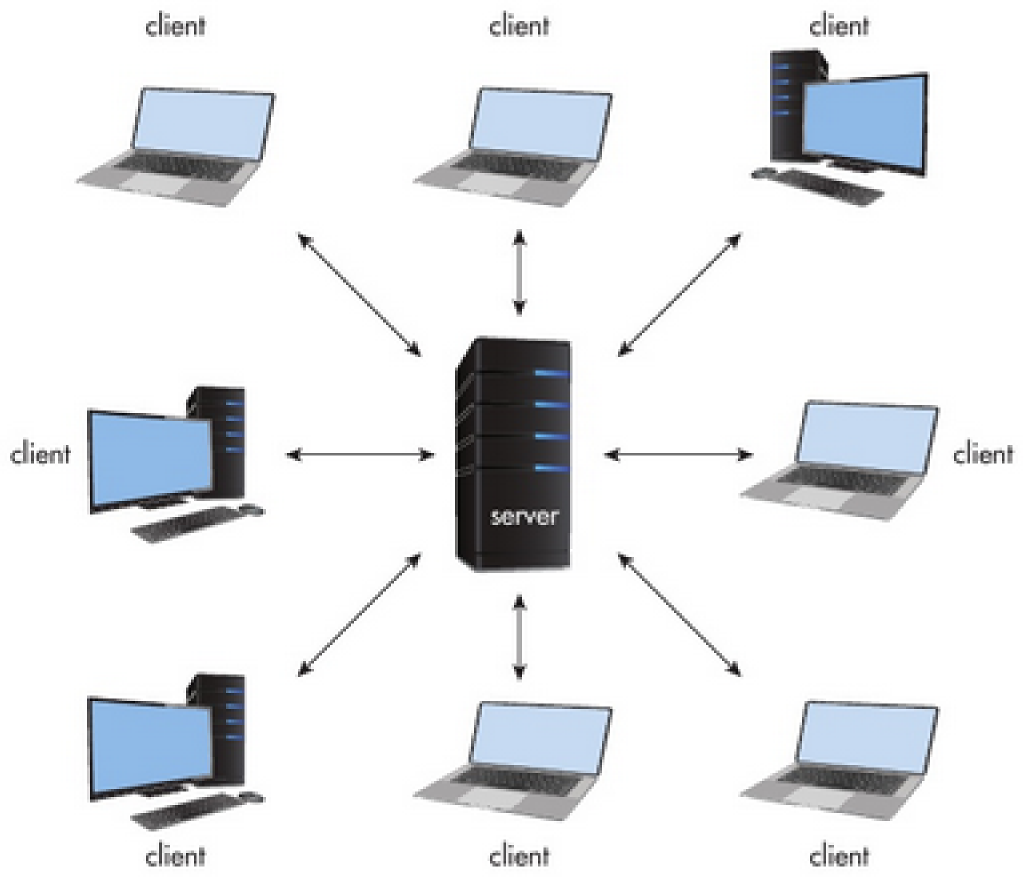


Figure 2.10 A client–server network

It is possible to set up very large networks using a client–server structure. The large amount of traffic that would be created by all the clients on the network can be managed by the different types of servers.

KEY TERMS

peer: a single computer in a peer-to-peer network

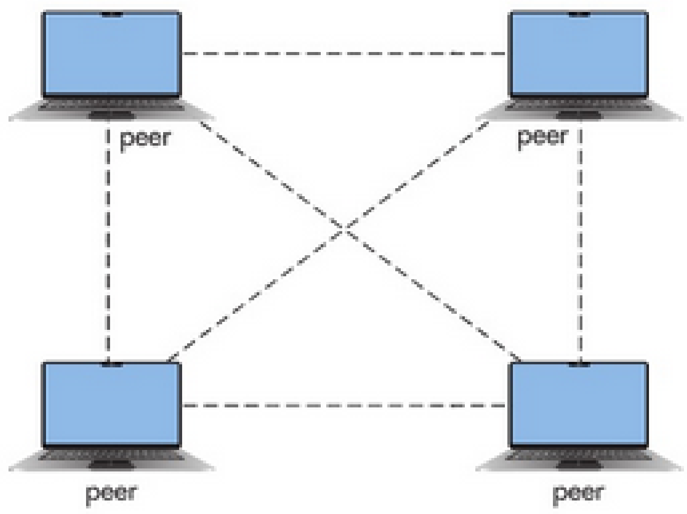
peer-to-peer network: a type of network that does not have a server as a point of control; all devices on a peer-to-peer network are equal

PEER-TO-PEER NETWORK

There is just one type of device in a **peer-to-peer network** and this is called a **peer** computer. This type of network can be as simple as one device connected to another, for example a mobile phone connected to a computer. A peer-to-peer network does not have a controlling computer at the centre. Each peer computer in the network has the same level of importance. The main reason that peer-to-peer networks are created is to share data. Table 2.3 shows some benefits and drawbacks of a peer-to-peer network.

TABLE 2.3: BENEFITS AND DRAWBACKS OF A PEER-TO-PEER NETWORK

Benefits	Drawbacks
Each peer computer can share files and resources. For example, they can share any files that they want others to see, and they can share resources such as a printer. To do this, they just need to set the files or printer to be public. Users can choose which files and resources they want to share and which they want to keep private.	The network is less secure than a client–server network, as each user is responsible for the security of their own computer and data. A user that is not as cautious with this may affect others on the network.
It is cheap to set up as it just needs transmission media to connect the computers. No expensive servers need to be purchased.	Each user is responsible for backing up their own data.
The network does not require any specialist technicians and is therefore also cheap to maintain.	If too many peer computers are added to the network, the volume of network traffic can slow down the network.
If one peer computer fails, it will only be the public files and resources stored on that computer that will be unavailable to other users.	



▶ Figure 2.11 A peer-to-peer network

Only small networks are set up using a peer-to-peer structure because the volume of traffic from a large number of devices would cause too many data collisions, which would reduce the performance of the network.

SKILLS

INTERPRETATION, CREATIVITY

ACTIVITY 3

Create an infographic that could be displayed in your classroom to show the difference between a client–server and a peer-to-peer network.

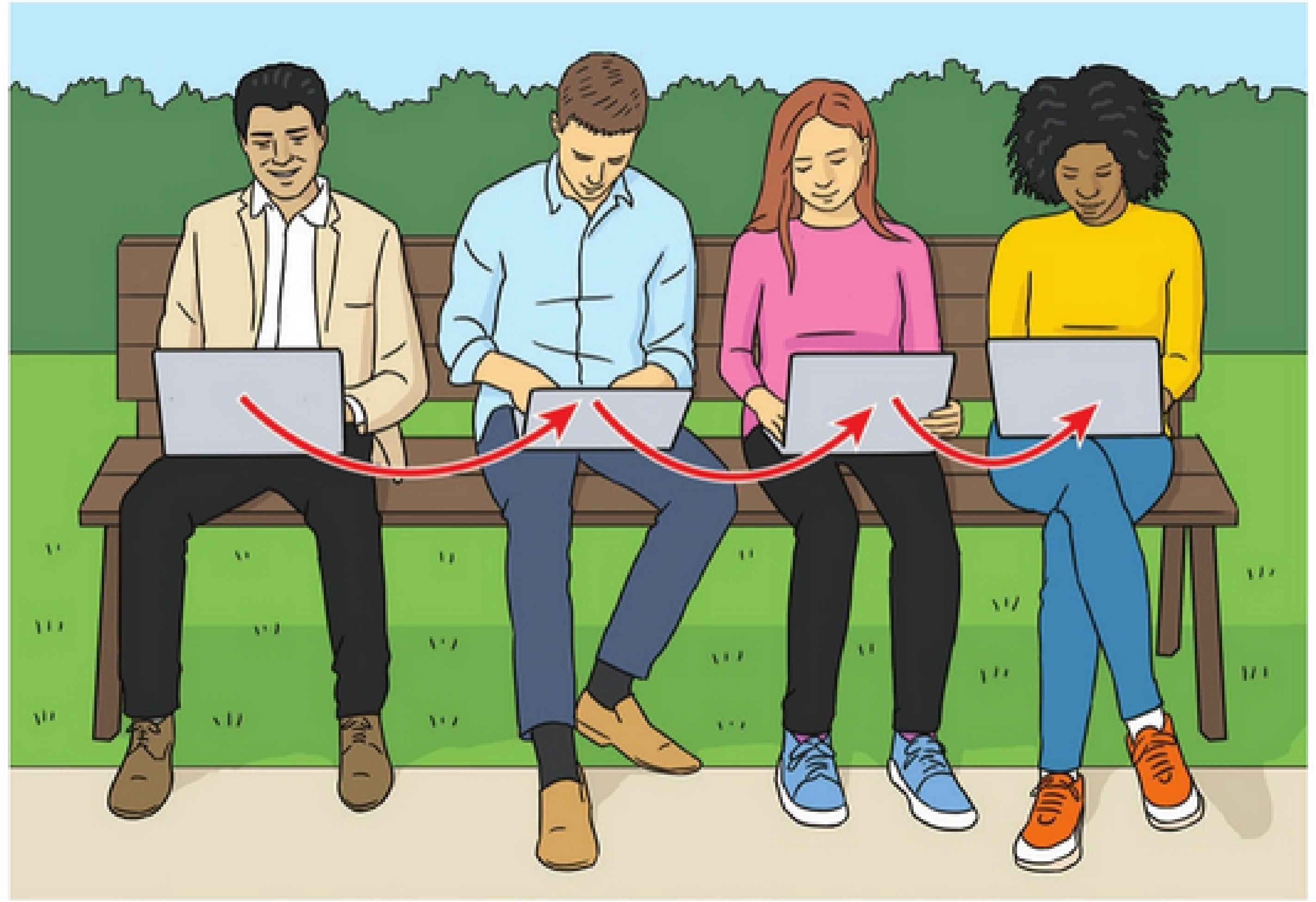
AD HOC NETWORK

KEY TERM

ad hoc network: a temporary network that is set up to share resources

An **ad hoc network** is a network that has been set up spontaneously for a temporary period of time. It has the same structure as a peer-to-peer network – there is no central point of control in the network. Ad hoc networks are normally set up to temporarily share resources, such as files.

This type of network is often used when network components such as a router are not available or in close enough proximity. For example, there might be a group of students working on a project together in a park. Each student in the group has a laptop, and they want to share the files they are creating for their project. There is no router or cables to connect their devices. One student decides to set up an ad hoc network to allow the other students to connect their laptops. This student sets up the network by transmitting their network ID and listening for devices that want to connect. All devices that connect to the ad hoc network can share data without the use of a router. Once the students have finished their work session in the park, the network is dismantled, as it was only needed for the set period of time.



▶ Figure 2.12 An ad hoc network allows users to temporarily share resources

Table 2.4 shows some benefits and drawbacks of an ad hoc network.

TABLE 2.4: BENEFITS AND DRAWBACKS OF AN AD-HOC NETWORK

Benefits	Drawbacks
The network can be set up anywhere as it does not require any special components.	Security can be weaker in this type of network. This is because the connection is created wirelessly and anyone close by could try to connect.
It is cheap to set up the network as no new network components need to be purchased.	If the ad hoc network becomes too large it could suffer issues such as data collisions and data loss because of the volume of network traffic.
	Not all devices are able to connect to an ad hoc network. Some portable devices such as mobile phones may not have a setting to allow them to do this.

KEY TERMS

hotspot: what a mobile phone becomes when it allows devices to be tethered to it to use its internet connection

tethering: the process of connecting another device to a mobile phone to make use of the phone's internet connection

Wi-Fi: a form of wireless access to the internet

SKILLS

ANALYSIS, CREATIVITY

ACTIVITY 4

Use the internet to find out how ad hoc networks are used in natural disaster zones.

TETHERING

Tethering is the process of connecting another device to a mobile phone to make use of its internet connection. Tethering a device to a phone to use its internet connection turns the phone into a **hotspot**. A hotspot can also be considered as a portable router. Tethering is mostly created using a **Wi-Fi** connection. It can also be created using Bluetooth or USB. You will learn more about Wi-Fi and Bluetooth connections later in this topic.

This type of network is usually a small network consisting of two devices, a mobile phone and another device such as a laptop or a tablet. A user will often tether their laptop or tablet to their phone to use its internet connection while travelling, for example to watch a movie on a larger screen than the one on their phone. Users may also choose to tether their laptop to their phone so that they can obtain documents emailed to them and edit them on a larger screen when they do not have a Wi-Fi connection.

▶ Figure 2.13 A laptop tethered to a mobile phone

Table 2.5 shows some benefits and drawbacks of tethering.

TABLE 2.5: BENEFITS AND DRAWBACKS OF TETHERING

Benefits	Drawbacks
Any device can be given an internet connection, even when a Wi-Fi network is not available.	Tethering can drain the battery of the devices, especially the mobile phone. If there is no ability to charge the device, the connection is not likely to last a long time.
Tethering means that multiple data plans are not needed for each device a user has. This can save on costs as only the mobile phone needs a data plan.	Tethering will use the data in the data plan for the mobile phone. Streaming video or downloading large files could use up a lot of the data allowance in the data plan in a short period of time.
Tethering can be more secure than using a public Wi-Fi connection, as the network created is a private network.	It is essential to set a strong password for the tethered connection. If not, it could be easy for someone to hack the tethered connection and access a person's data.

NETWORK METRICS

PERFORMANCE

▶ Network metrics are ways that the performance of a network can be measured

The performance of a network can be measured in several different ways. These are called the network metrics. Table 2.6 shows the network metrics that you need to know about.

KEY TERMS

bandwidth: the maximum amount of data that can be transferred from one device to another in a set amount of time

speed: the speed at which the data is transferred from one device to another

throughput: the amount of data that actually passes through the network from one device to another in a set amount of time

TABLE 2.6: THE MAIN NETWORK METRICS

Network metric	Description
Speed	One of the most used network metrics. The network speed is simply the speed at which the data is transferred from one device to another. It is usually measured in bits per second (bps) or sometimes bytes per second (Bps).
Bandwidth	The maximum amount of data that can be transferred from one device to another in a set amount of time. This is also usually measured in bps or Bps and is the main metric that affects the speed of the network. For example, a network that has a bandwidth of 50 Mbps can transfer up to 50 million bits of data per second. A higher bandwidth will normally mean faster speeds for the network. The available bandwidth of a network is shared between the devices that are part of the network. This means that the more devices there are on a network, the more the bandwidth will need to be shared, which will make data transfer speeds slower.
Throughput	The volume of data that actually passes through the network from one device to another in a set period of time. The bandwidth is the maximum speed of the network, which is the potential volume of data that can travel from one device to another. The throughput is the actual speed of the network, which is the actual volume of data that travels between devices in a set period of time. Throughput is also measured in bps.

(Continued)

KEY TERMS

availability: a measure of the amount of time a network is available for use

error rate: a measure of the number of errors that occur in a network

jitter: a measure of the disruption that occurs when data is transmitted across a network

latency: the delay that can be present in the transmission of data across the network

packet loss: a measure of the number of packets that were transmitted from the sending device but did not arrive at the receiving device

scalability: how well the capacity of a network can be increased or decreased to respond to different amounts of data transfer

TABLE 2.6: THE MAIN NETWORK METRICS (*Continued*)

Network metric	Description
Scalability	A network's ability to be able to handle a larger volume of data than it normally handles. One way that a network can be made scalable is by being able to increase its bandwidth when necessary. For example, if an online company has a sale at 50% off for their products, they will need to scale their network to be able to handle the amount of extra traffic they will potentially get from all the customers who take advantage of the sale.
Latency	The delay in the transmission of data across the network. Latency is measured using the time taken for data to make a round trip in the network. This means how long it takes to go from the sender to the receiver and back to the sender again. For example, if a user sends a request to a web server for a web page, the web page is then returned to the user. The time taken between the request sent and the web page returned is called latency. A network with low latency times has a higher performance than a network with high latency times.
Error rate	A measure of the number of errors that occur in a network. It is usually measured as a percentage. The types of errors measured include data packet loss, data packet corruption and data collisions. A network with a high error rate can cause problems, as each data packet that does not arrive correctly will need to be re-sent. This will create more traffic and reduce the performance of the network.
Packet loss	A type of error that can occur in a network. It is a measure of the percentage of transmitted packets that were not received at the receiving device. Packet loss is also normally measured as a percentage. A packet loss of just 1% can affect the performance of a network.
Availability	A measure of the amount of time a network is available for use. It is calculated as the percentage of time a network is actually available out of the total time it should have been available. The network may have been unavailable due to power outages, hardware issues or downtime to update the network. Network availability is normally measured as a percentage.
Jitter	A measure of the disruption that occurs when data is transmitted across a network. The disruption measured is often caused by network congestion. If data sent across a network gets disrupted because of latency or network errors, the network can start to lose data packets. The process of this disruption is the jitter of the network.

SKILLS

ANALYSIS, CREATIVITY

ACTIVITY 5

Use the internet to:

- do a speed test on your computer
- find out how throughput is calculated.

Sometimes a network will need to be scaled. For example, a company might double in size, and so need twice as many employees. Or it may open another office in a different location, or move to a hybrid working model where some employees work remotely for some or all of the working week.

Use the internet to research how a network could be scaled to cater for these changing demands.

NETWORK PROTOCOLS

COMMUNICATION PROTOCOLS

KEY TERM

protocol: a set of rules that govern the transmission of data in a network

You have learnt a lot about the components that are used in a network, how a network can be structured and how the performance of a network can be measured. You will now learn about the different ways in which devices can communicate using different network **protocols**. A protocol is a set of rules that govern the transmission of data in a network. There are a wide range of protocols that you need to understand. Most of these protocols are regulated by an organisation called the Institute of Electrical and Electronics Engineers (IEEE). The first set of protocols are communication protocols:

- ethernet
- Wi-Fi
- Bluetooth
- ZigBee®
- cellular
- infrared.

Ethernet

Ethernet is a protocol used to connect devices in a LAN using a wired connection. An ethernet cable is used to connect devices that use the ethernet protocol. Most of the wired connections that are created in a network are done using an ethernet cable. Ethernet cables can be coaxial, twisted pair or fibre optic.

Ethernet was invented in 1973 by an American engineer named Robert Metcalfe. He invented it to develop a set of rules to allow devices to be connected using a cable so that they could communicate with each other. Ethernet could initially transfer data at an average of 3 Mbps. It has been developed and improved over the years and can now reach speeds of data transfer up to 100 Gbps.

▲ A twisted pair ethernet cable

Ethernet is still used to connect many devices in organisations, because it is:

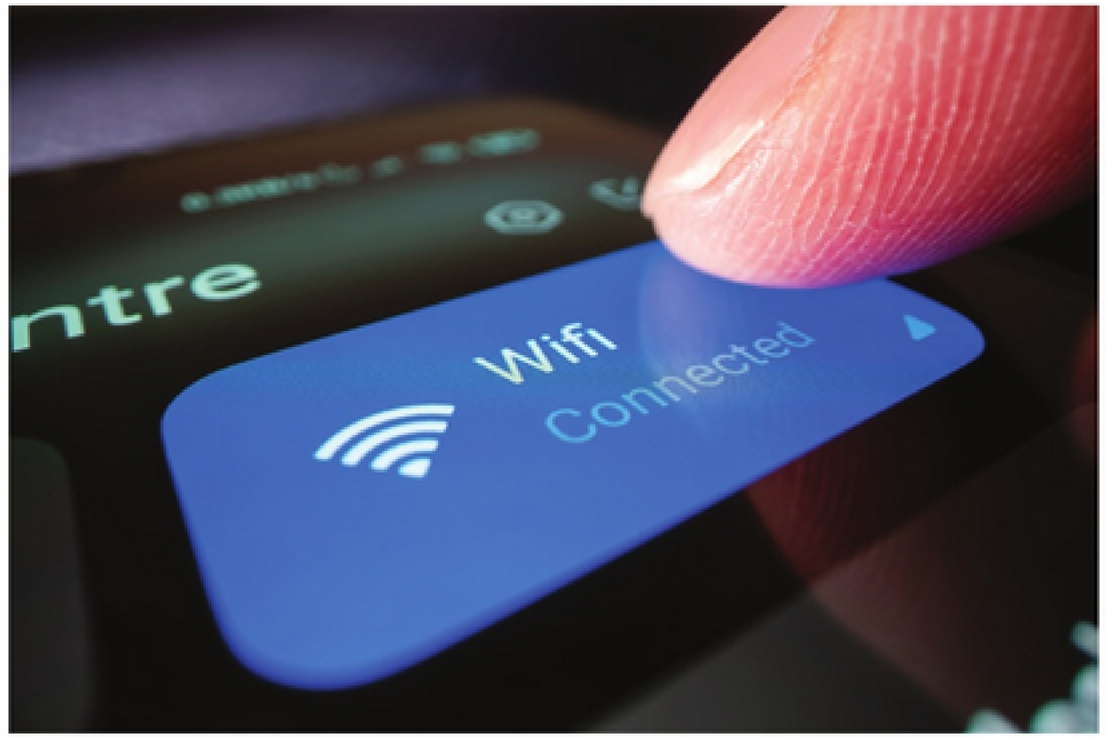
- cheap to buy (low manufacturing costs)
- more secure than a wireless connection
- reliable; it uses error detection methods to detect any errors in data transmission
- compatible with many devices that require a wired connection to a network; many devices such as computers and routers are built with a port for an ethernet cable
- easy to install; it only requires an ethernet cable to be plugged into an ethernet port.

Wi-Fi

KEY TERM

Wi-Fi: a type of wireless data transmission that uses radio waves to transmit data

Have you ever wirelessly connected your laptop to the router in your home? If you have, you have used a **Wi-Fi** connection to do this. Wi-Fi is a protocol that is used to transmit data wirelessly between two devices. Wi-Fi transmits data using radio waves, and it can transmit data using two different frequencies – 2.4 GHz and 5 GHz. Currently 2.4 Ghz is the most common frequency in use. However, this frequency is also used by several other devices, so it can often have a lot of traffic. The 5 GHz frequency was developed partly for this reason, but also because it allows faster transfer of data.



▲ A Wi-Fi connection can be made from devices such as smartphones

Wi-Fi was first developed in 1997 and could initially transfer data with speeds up to 2 Mbps using the 2.4 GHz frequency. The 5 GHz frequency was developed in 2009, and transfer speeds increased up to speeds of 20 Gbps. By 2023, six versions of the Wi-Fi protocol had been released, meaning Wi-Fi can now transfer data at speeds up to 10 Gbps. Wi-Fi using the 2.4 GHz frequency can reach up to 45 m. Wi-Fi using the 5 GHz frequency can reach up to 15 m.

Bluetooth

Have you ever connected wireless headphones to your mobile phone? If you have, you have probably used a **Bluetooth** connection to do this. Bluetooth is a protocol that is used to transmit data wirelessly between two devices. Bluetooth also transmits data using radio waves. Bluetooth only uses the 2.4 GHz frequency to transmit data.

KEY TERMS

Bluetooth: a type of wireless data transmission that uses radio waves to transmit data between two devices that are in close range (within 10 m)

pairing: the name given to the connection made between two devices that are using a Bluetooth connection

Bluetooth was invented in 1994, but it was not used in a commercial product until 2000, when the company Ericsson launched a wireless Bluetooth headset. To create a Bluetooth connection, devices need to be connected using a system called **pairing**. When a user wants to connect two devices, for example a smartphone and a pair of wireless headphones, they will do this using the host device. In this case it will be the smartphone. They will use the Bluetooth setting on the phone to scan for devices that allow a Bluetooth connection. A list of compatible devices will be shown on the phone and the user can select the one they want, in this case the wireless headphones. The Bluetooth connection can then be created between the phone and the wireless headphones. Sometimes this can require entering a PIN code into the phone. You can connect up to seven different devices to the host device, but each device must be within 10 m of the host device. A Bluetooth connection can transfer data at speeds of up to 3 Mbps.

▶ A smart watch and a smartphone can be paired using Bluetooth

ZigBee®

KEY TERM

ZigBee®: a type of wireless data transmission that uses radio waves to transmit data between two devices; it is one of the main methods of connection used in the internet of things (IoT)

Have you ever used your voice to command a virtual assistant (e.g. Amazon Alexa) to turn on the lights in your home, or change the temperature of a room? If so, you have used a **ZigBee®** connection to do this. ZigBee® is a protocol that is used to transmit data wirelessly between devices. ZigBee® transmits data using radio waves using the 2.4 Ghz frequency.

ZigBee® is a very similar protocol to Bluetooth®, but is normally used to connect different types of devices. Bluetooth is used to connect devices such as smartphones and wireless headphones, or smartphones to a car entertainment system. ZigBee® is used to connect devices such as smart light bulbs to virtual assistant technology. ZigBee® connections are the basis of the IoT. To create a ZigBee® network in a building, a ZigBee® hub is needed. All devices that can use ZigBee® will be connected to the ZigBee® hub to allow them to communicate. ZigBee® can transfer data at speeds up to 250 Kbps, and it has a range of up to 100 m.

▶ A touchscreen home control system uses ZigBee® to control heating and lights

SKILLS

ANALYSIS, CREATIVITY

ACTIVITY 6

Create a table that shows a comparison between the wireless connections of Wi-Fi, Bluetooth and ZigBee®. You could choose key features of the connections and add descriptions for each. For example:

Feature	Wi-Fi	Bluetooth	ZigBee®
Transmission method	Radio waves	Radio waves	Radio waves
Radio frequency			
Maximum data transfer speed			

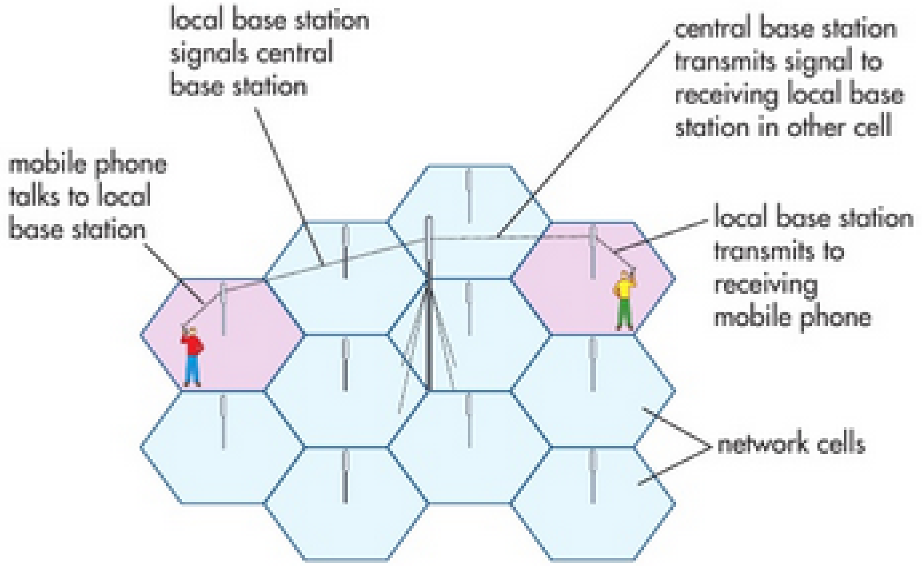
KEY TERMS

base station: a large tower that is used to transmit data across a cellular network

cellular network: a type of network made up of cells that each have a base station that is used to transmit signals used by mobile phones

Cellular

Have you ever called a friend or a family member using a mobile phone? If you have, you will have used a **cellular network**. A cellular network is made up of lots of cells that cover large geographic areas, for example a country. Each cell uses some hardware called a **base station**. The cells in a cellular network are arranged in a honeycomb structure.


local base station signals central base station
central base station transmits signal to receiving local base station in other cell
mobile phone talks to local base station
local base station transmits to receiving mobile phone
network cells

▶ Figure 2.14 A cellular network structure

A cellular network is what mobile phone users connect to when they want to carry out actions on their phone, such as make a call. Both radio waves and microwaves are used to transmit data in a cellular network. Radio waves are normally used to transmit data from a phone to a base station and microwaves are often used to transmit data from one base station to another.

When a user calls their friend using a phone, the data is transmitted to the base station in the cell that the user is currently placed in. If their friend is within the same cell, then the data is transmitted from that same base station to the friend's phone. If the friend is within a different cell, the data is sent from base station to base station until it reaches the base station in the cell where the friend is. It is then transmitted from this base station to the friend's phone.

The global system for mobile communications (GSM) is a set of protocols that are used to transmit data across cellular networks. These are the rules that govern how the data is sent. These protocols can also be known by the term 2G (2nd generation) network. These protocols have been expanded upon and improved to first create the 3G network, then the 4G network, and most recently the 5G network. The different generations of networks were developed to improve the data transfer rate:

2G has a data transfer rate of speeds up to 0.1 Mbps

3G has a data transfer rate of speeds up to 42 Mbps

4G has a data transfer rate of speeds up to 150 Mbps

5G has a data transfer rate of speeds up to 10 Gbps.

SKILLS

ANALYSIS, CREATIVITY

ACTIVITY 7

Use the internet to find out why mobile phone calls sometimes cut off mid-conversation. What is happening in the cellular network for this to occur?

Infrared

You learnt about how data is transmitted using infrared earlier in this topic. The protocol used in infrared communication is the national electrical code (NEC) infrared protocol.

SKILLS

ANALYSIS, CREATIVITY

ACTIVITY 8

Tariq has a home network and wants to add some new smart devices.

Tariq wants:

- the two existing PCs in the upstairs office in his house to remain connected using a wireless connection
- the existing laptop to be connected using Wi-Fi
- a WAP to allow any new devices that have wireless capabilities to be connected
- the new smart bulbs to be connected using ZigBee®
- the new smart refrigerator to be connected using ZigBee®
- a powerline connection between the router in the downstairs lounge and the PCs in the upstairs office.

Draw a diagram to show a network design that will meet Tariq's requirements.

NETWORK PROTOCOLS

There is a further set of protocols that are used to govern the transmission of data across a network:

open systems interconnection (OSI) model

transmission control protocol/internet protocol (TCP/IP)

voice over internet protocol (VOIP)

session initiation protocol (SIP).

OSI model

The **OSI model** is a framework that was developed in 1984. It is a theoretical framework that can be used as the basis for structuring the transmission of data from one computer to another across a network. The OSI model divides the protocol into 'layers', with each having a specific function. Data is passed from one layer to the next. Many other protocols are based on some of the layers of the OSI model. Table 2.7 shows the seven layers of the OSI model.

KEY TERM

OSI model: a theoretical model that is the basis of many protocols

TABLE 2.7: THE OSI MODEL

Layer	Description
Application (7)	This is the top layer in the framework. This layer directly interacts with data from a user. It receives the data from the user and displays the data for the user. Software applications such as a word processor are not part of the application layer. However, software such as web browsers and email clients are part of the application layer. The application layer provides several protocols that are used in the transmission of data, including the hypertext transfer protocol (HTTP) and the file transfer protocol (FTP).
Presentation (6)	This layer of the framework prepares data for use in the application layer. It makes sure that data is in the correct format for the intended device. This could involve encoding, encrypting or compressing the data.
Session (5)	The role of this layer of the framework is to create channels of communication (called sessions) between devices. It is responsible for opening a session, making sure it's open for long enough to transmit the data, then closing the session.
Transport (4)	This layer takes the data that is transmitted in the session layer and breaks it down into segments to prepare it for transmission, and it also reassembles the segments after the data has been transmitted. The transport layer is also responsible for sending the data at the speed that the network is prepared for. It also checks the data for errors and requests any data be re-sent when errors are found.
Network (3)	The role of this layer is to take the segments of data from the transport layer and break them down into smaller units called packets of data. It reassembles received packets back into the segments required by the transport layer. This layer is also responsible for finding the best route for the packets of data to travel across the networks. It normally does this using a router.
Data link (2)	The role of this layer is similar to the network layer. However, the network layer is normally used when transmitting data between different networks, for example across a LAN and a WAN. The data link layer is used when transmitting data from one device to another within a single network, for example a LAN. To do this it takes the packets of data created in the network layer and converts them into frames. The frames are transmitted from the sending device to the receiving device. This layer has two sublayers: • logical link control – manages protocols and handles error checking • media access control – uses MAC addresses to connect devices.
Physical (1)	This layer is all the wired cables or wireless connections that are made between the devices in the network. The data is broken down into bits for the layer so that it can be transmitted down a cable or across radio waves.

EXAM TIP

There is a quick phrase you can say to remember the first letter of each of the layers.

All **P**eople **S**eem **T**o **N**eed **D**ata **P**rocessing

All – Application

People – Presentation

Seem – Session

To – Transport

Need – Network

Data – Data link

Processing – Physical

KEY TERM

TCP/IP model: a protocol that is used to transmit data across the internet

TCP/IP model

The **TCP/IP model** is older than the OSI model. It was developed by the US Department of Defense in the 1960s. TCP/IP is a protocol that has four layers rather than the seven layers present in the OSI model. Table 2.8 shows the four layers of TCP/IP.

TABLE 2.8: THE TCP/IP PROTOCOL

Layer	Description
Application layer	This layer incorporates the role of the application, presentation and session layers from the OSI model.
Transport layer	This layer is the same as the transport layer in the OSI model. It uses the transmission control protocol (TCP).
Network/internet layer	This layer is the same as the network layer in the OSI model. It uses the internet protocol (IP).
Link/physical layer	This layer incorporates the role of the data link layer and the physical layer from the OSI model.

TCP/IP is the main protocol used by devices communicating on the internet. Each time data is sent from one device to another over the internet, the TCP/IP model is used to prepare and send the data.

SKILLS
ANALYSIS

ACTIVITY 9

Use the internet to find out which protocols are used in the application layer of TCP/IP.

KEY TERMS

SIP: a protocol that is used as part of VOIP

VOIP: a protocol that is used to send data when one person is making a voice call to another person using the internet

VOIP and SIP

VOIP is a protocol that is used to send data when one person is making a voice call to another person using the internet. VOIP was invented in 1995 and was developed to allow people to use devices other than their mobile phone to make a call. People use VOIP to make calls on devices such as a PC or laptop. VOIP is responsible for converting a person's voice into digital data, compressing it for transmission and then transmitting the data across the internet.

VOIP involves the use of another protocol called **SIP**. SIP has the same responsibilities as a protocol in the application layer of the OSI model.

▶ Applications like Skype and FaceTime use VOIP

NETWORK DESIGN

ADDRESSING IN A NETWORK

If someone sends you a letter in the post, how does that letter get to you? How does the postal service or delivery company know where to send it? They use your address! Each device that is part of a network also has an address. They actually have two different addresses. These addresses are used for the same purpose as your address at home – to make sure that information can be sent to the correct destination. The two addresses are:

- a MAC address
- an IP address.

MAC address

A MAC address can be used to identify a device in a network. The MAC address is assigned by the manufacturer of the device when it is being made. The MAC address is actually given to the NIC that is built into the device.

A MAC address has several features.

- Each address is unique.
- It is written in hexadecimal.
- It has 12 digits and the digits are grouped in pairs.
- Each pair of digits is separated by a colon.
- The first three pairs of digits are the manufacturer's ID.
- The second three pairs are the serial number of the device.

An example of a MAC address is:

3C:5A:B4:01:63:D2

A MAC address is normally used in a LAN when data is sent from one device to another within the LAN. The MAC addresses of the sending device and the receiving device are included in every packet of data that is transmitted across the network. You learnt earlier in this topic that it is a switch in a network that reads the packet of data to find the receiver's MAC address and forwards the packet on to this device.

SKILLS
ANALYSIS

ACTIVITY 10

Find out what the MAC address is of the computer you are currently using. If you are not using a computer at the moment, you could try and find out the MAC address of another device, such as your smartphone.

You can use the internet to find out how to locate the MAC address of the computer or device.

IP address

An IP address can also be used to identify a device in network. An IP address is normally automatically assigned by a router in a home network when a device is connected. The router normally does this using the **dynamic host configuration protocol (DHCP).** When a device connects to a network, a request is sent to the router to assign the device a unique IP address. The DHCP service on the router manages this process. In the networks of large organisations, for example a company, university or school, there will usually be a dedicated DHCP server. This is done so that the router can concentrate on what it does best, which is to route data packets. An IP address can also be assigned by an internet service provider (ISP). This is the company that provides a user's internet connection.

KEY TERM

dynamic host configuration protocol (DHCP): a protocol that is used to assign an IP address to a device when it is connected to a network

An IP address has several features.

- Each address is unique to the network that the device is connected to.
- It can be static or dynamic.
  - A static address means the device will always have the same IP address.
  - A dynamic address means the IP address of a device can change each time it connects to a network.
- There are two different versions of IP addresses – IPv4 and IPv6. Table 2.9 shows the differences between them.

TABLE 2.9: FEATURES OF IPV4 AND IPV6

Features of the IPv4 format	Features of the IPv6 format
32-bit address	128-bit address
Normally represented as denary	Normally represented as hexadecimal
Has four sets of digits	Has eight sets of digits
Each set of digits is separated by dots	Each set of digits is separated by colons
	A set of four 0 digits can be replaced by a double colon ::
Example address: 128.11.32.40	Example address: AB01:EF01:2368:BB30:4324:10DB:0455:BA10 or AB10:F1B1::AF00:3121::8998:FF22

SKILLS
ANALYSIS

ACTIVITY 11

Find out what the IP address is of the computer you are currently using. If you are not using a computer at the moment, you could try and find out the IP address of another device, such as your smartphone. Does the IP address look like an IPv4 or IPv6 address? How do you know?

You can use the internet to find out how to locate the IP address of the computer or device.

NETWORK SECURITY

When a device is connected to a network, the device and all the data stored on the device become vulnerable. The data may be valuable, and there may be people who will try and steal it or corrupt it for different reasons, such as:

- fraud
- identity theft
- intellectual property theft
- revenge
- ethical protests
- entertainment value.

KEY TERMS

authentication: a collection of security methods that are used to verify the identity of a user when they are trying to access a device or an account

hack: to gain unauthorised access to data

malware: a type of malicious software that is used to damage data or a device

The two main threats that a device can be exposed to when connected to a network are:

- people who want to **hack** the device to steal the data stored on it
- people who want to install **malware** on the device to corrupt the data stored on it.

A user's device is particularly vulnerable when it is connected to an open network. This is a network that is not secured and it can be used by anyone that is in the local area. This type of network is normally a wireless network, and they are often found in places like shopping malls and restaurants.

SECURITY SOLUTIONS

Security solutions can be put in place by a user or an organisation to help stop a device being so vulnerable when connected to a network.

Authentication

One security solution that can be used is **authentication**. This is designed to verify the identity of a user when they are trying to access a device, or an account that the user may have, such as a social media account or a bank account. Table 2.10 shows the three main types of authentication.

TABLE 2.10: TYPES OF AUTHENTICATION

Authentication type	Description
Password	This is a series of characters that a user needs to enter to be able to access the device or the account. The user will initially set the password and then each time they want to log in they will need to re-enter the password. Users are normally encouraged to set strong passwords as they are more secure. This is because they are harder for a hacker to guess. A strong password is one that is long and complex, and often contains letters, numbers and symbols.
Two-factor	This is a method that requires a second verification from the user. This is normally in the form of a numeric code that is sent to a user's device. The user will log in to their account using a password, at which point a two-factor authentication check is triggered. A code is sent to the user's device, for example their mobile phone, and they then enter this code to gain access to their account. This makes access to the account more secure, as a hacker would need to have the user's phone to get the code. Two-factor authentication is frequently used for online banking.
Biometrics	This is a method that requires a user's biological data to act as a password. Instead of a character-based password, a person can use biological data like their fingerprint or facial features as a password. This makes access more secure as it is very difficult to fake a person's biological data. Some common examples of biometrics are a fingerprint scanner, iris scanner, facial recognition software and voice recognition software. You may use a fingerprint scanner or facial recognition software to unlock your phone.

▶ Biometric security devices can use data such as facial features, a fingerprint or a person's voice as a password

KEY TERMS

access rights: restrictions that are placed on who can see and edit data

anti-malware: software that scans a device and removes or quarantines malware

security settings: ways that a user can control who can see or access data that they store in an account

Security settings

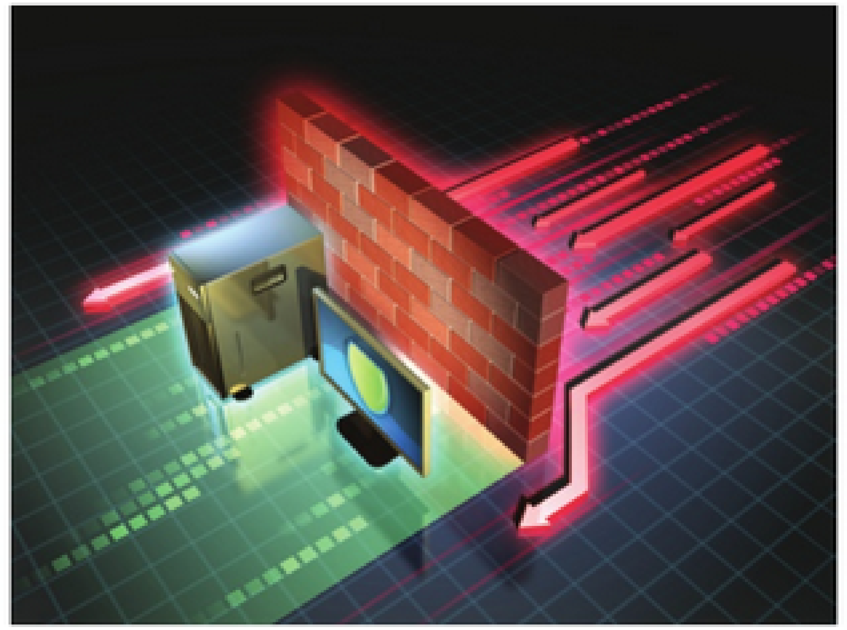
Another security solution that can be used is to review the **security settings** of a device or account. For example, a social media account will allow you to change your security settings. You can change settings such as making more of your data private so that it cannot be seen by other users. You can also change settings like adding two-factor authentication to your account.

Access rights

Access rights are not designed to directly stop a hacker accessing an account. They are designed to limit the data that the hacker can see. They are also designed to limit the data seen by other people in an organisation, to make sure that people only see the data that is relevant for their work. Access rights are normally attached to a person's username. When the user logs in to their account, the username will show the network server what data the user should be allowed to see and what they can do with that data. Some people may only be able to read the data. Some people may be able to also edit the data. For example, the headteacher in a school may have different access rights for students' data from the other teachers in the school. The headteacher may be able to amend any student's data, but other teachers can only see the student's data. This keeps the data more secure as it means it is less likely to be accidentally deleted or corrupted by a person. It also means that if a hacker does manage to hack into a person's account, they may not be able to see all the data stored by the organisation.

Anti-malware

Another security solution that can be used is **anti-malware**. One of the main threats of connecting to a network is having malware installed onto your device. Malware is malicious software designed to damage your device or the data stored on it. Common examples of malware are viruses, spyware, trojan horses and ransomware. Malware is normally installed onto a device when a user unknowingly downloads a malware-infected file from the internet. Anti-malware is used to scan a device for malware. If it finds a file that it thinks is malware installed onto the device, it can either delete it or quarantine it for the user to review and delete. Some anti-malware can also scan a file that the user wants to download. It can check to see if there is any malware present in the file before the user downloads it. Anti-malware can help keep a device more secure by protecting it from any malware threats. It is important to regularly update any anti-malware that you use. This is because new malware is being created all the time, so a regular update will inform your anti-malware software about any new known malware.



▲ A firewall can help keep a computer safe by examining incoming and outgoing traffic

Firewall

An additional security solution that can be added is a **firewall**. A firewall can be hardware based or software based. A hardware firewall is a separate device that is placed between the computer and the network. A software firewall is software that is installed onto the computer. A firewall acts like a security guard for data. A firewall examines all the data traffic coming into and leaving a computer or network. It examines the traffic against a set of criteria. These criteria can be set and changed by a user or network manager. The criteria are a set of rules for the traffic. The firewall will accept the traffic and let it through or block the traffic, depending on whether it meets or does not meet the set criteria. An example of the type of criteria that can be set is a blacklist and a whitelist of IP addresses. The blacklist is for IP addresses that are not allowed to send data to the network or devices in the network that are not allowed to send data to those addresses. A whitelist is for IP addresses that are allowed.

KEY TERMS

encryption: a method of security that scrambles data to make it meaningless

firewall: a type of security setting that can be used to examine data incoming and outgoing from a computer or network

physical controls: physical security measures that can be used to keep data secure, such as CCTV and locks on doors

LINKS

Symmetric encryption is covered in more detail in Topic 13 in Student Book 2.

A firewall helps to protect a device from hackers and malware as these are two things that can be recognised from the criteria set for a firewall. A hardware firewall will examine all the data before it reaches the computer. This is generally thought to be the safer method as it means that no malicious traffic should actually get into the computer. A software firewall will allow malicious data to enter the computer, but it will recognise it as malicious and dismiss it from the computer.

Physical controls

A collection of security methods that are often used in workplaces such as an office are called **physical controls**. Physical controls are things such as locks on doors, alarm systems, keycard entry systems and CCTV cameras. Physical controls help keep devices and data secure by acting as a deterrent for anyone who tries to steal the device or gain access to the data.

Encryption

A final method of security that can be used is **encryption**. This is when data is scrambled to make it meaningless. Data is encrypted using an algorithm called an encryption key. Encryption can be symmetric or asymmetric. Symmetric encryption uses the same key to encrypt and decrypt the data. Asymmetric encryption uses different keys to encrypt and decrypt the data. A simple symmetric encryption key could be that every letter in a text file is moved three places forwards in the alphabet. The sentence 'ICT is awesome' would then become 'LFW lv dzhvrph'. Although encryption cannot stop data being stolen, it keeps data more secure by making sure that if it is stolen by a hacker, the hacker will not be able to understand the data as it will be meaningless.

SKILLS
CREATIVITY

ACTIVITY 12

Create a storyboard for an advert that could be displayed on a social media website informing users of three different ways that they can help keep their data safe.

If you are not sure how to create a storyboard, you can search online for websites that can help you with this, such as www.boards.com.

THINKING LIKE AN IT SPECIALIST

Almost all companies use some kind of network to send and store data. These networks can vary a lot in size and in the components that they contain.

How do you think a company that runs a nationwide chain of clothes shops can decide what is needed in its network?

CHECKPOINT

1. What are two ways that devices can be connected using a wired connection?
2. What are two ways that devices can be connected using a wireless connection?
3. What is the role of a router in a network?
4. What is the role of a server in a network?
5. How many layers does the OSI model have?
6. What is the top layer of both the OSI model and the TCP/IP model?
7. What does it mean when a device is hacked?
8. What are two methods of authentication?
9. What is anti-malware used for?
10. What are two examples of physical controls that can be used to help keep a network or device secure?

SKILLS
ANALYSIS, INTERPRETATION

EXAM PRACTICE

1. A computer is connected to a network using a cable. The computer has an NIC. The network uses both a switch and router and it has a client–server network structure.

(a) State the role of the NIC. [1 mark]

(b) Explain the differences between a switch and router. [4 marks]

(c) The cable is a copper-based cable.

Which of these cables are copper-based? Write the correct letters in your exercise book. [1 mark]

A Twisted pair
B Fibre optic
C Powerline
D Coaxial

(d) Give two benefits of a client–server network structure. [2 marks]

2. A student tethers their mobile phone to their laptop.

(a) Explain what is meant by tethering. [2 marks]

(b) Give one drawback of tethering. [1 mark]

3. The table contains terms and descriptions of network metrics.

Copy and complete the table with the missing terms and descriptions. [4 marks]

Term	Description
	This is the maximum amount of data that can be transferred from one device to another in a set amount of time.
Latency	
	This is the amount of data that actually passes through the network from one device to another in a set amount of time.
Error rate	

4. The OSI model is a model on which protocols are based. It has seven different layers.

(a) Copy and complete the diagram to show the missing layers of the OSI model. The layers should be shown in order. [4 marks]

Application
Transport
Physical

(b) Describe the role of the transport layer in the OSI model. [3 marks]

5. An IP address can have an IPv4 or an IPv6 format.

Explain three differences between an IPv4 and IPv6 format. [6 marks]

6. Explain how a firewall can be used to help keep data that is stored on a computer safe. [4 marks]

TOPIC 3 THE ONLINE ENVIRONMENT

Many of us use the internet and the world wide web on a daily basis, but what do we really know about them? In this topic you will learn about different types of web pages and the scripting methods that are used to build them. You will also learn about online environments and online communities, and how they are used by people as places of work, places to share information and places to connect about shared interests. When we use the internet, we leave a digital footprint. You will learn how this happens and how we can make sure that we are responsible about the digital footprint that we leave. There are also risks to using the internet surrounding our personal data and identity, and some of these risks are linked to cloud computing. You will learn to recognise these risks, understand the impact they have and learn how to keep yourself safe when online.

TOPIC 3 THE ONLINE ENVIRONMENT

LEARNING OBJECTIVES

By the end of this topic you should be able to understand:

- the features and functions of the internet and the world wide web
- what is meant by static and dynamic web page content and how they are different
- the role of client-side and server-side scripting
- the impact of working in online environments
- the concept of online communities and the impact they have on an individual and an organisation, including a range of ways they can be used to make money
- the security risks of storing personal data online and how it can be protected
- the concept and impact of cloud storage and cloud computing
- what is meant by a digital footprint and the positive and negative effects of this.

GETTING STARTED

Make a list of all the personal data that you have ever shared online. Did you realise how much of your personal data you had shared? What might a person know or possibly infer about you from the personal data that you have shared?

PERSONAL INFORMATION

▲ Figure 3.1 How much personal data have you shared online?

KEY TERMS

internet: a network of computer systems and all the cables and other hardware that are used to connect them

world wide web: the collection of web pages and websites that are viewed using the internet

THE INTERNET AND THE WORLD WIDE WEB

We often use the **internet** and the **world wide web** to mean the same thing, which is the things that we see when we are online. However, the internet is all the infrastructure that is used to access web pages and send data across the world, whereas the world wide web is all the web pages that we look at when using the internet.

▲ Figure 3.2 The world wide web is a very large collection of web pages

KEY TERMS

domain name: a part of a URL that has the name of the web page; for example, the domain name for the URL https://www.pearson.com is pearson.com

domain name system (DNS): a type of server or service that stores a database of domain names and their matching IP addresses

uniform resource locator (URL): a text-based address for a website

web browser: software that is used to access web pages

web server: hardware that is used to store web pages and respond to requests for those web pages

All the web pages that we access when using the internet are stored on **web servers** around the world. Each web server has a unique IP address. Each web page has a **uniform resource locator (URL)**, for example https://www.pearson.com. A user enters the URL into their **web browser** to access a web page. The web browser needs to send a request to the web server for the web page, but first the URL must be converted to an IP address. To do this the web browser first sends a request to a **domain name system (DNS)** server. This is a computer that stores a list of **domain names** and their matching IP addresses. The IP address is the address of the web server where the web page is stored.

As there are over a billion different domain names, there are many DNS servers to store them all. The web browser extracts the domain name from the URL and sends this to a DNS server. The DNS server searches its database to see if it can find the domain name. If it can, it sends the matching IP address back to the web browser. If it cannot find the domain name, it sends it on to a DNS server at the next level so that it can search its database. The name request will keep on being sent up the DNS hierarchy as far as a top-level DNS server (at the 'root' level) until the name is found. If no result is found, a message will be sent back to the web browser to state that the web page could not be found. When the web browser has the IP address where the web page is stored, it uses this to send a request to the web server. The web server responds by sending the data for the web page back to the web browser.

SKILLS
CREATIVITY

ACTIVITY 1

Draw a diagram that shows how a web page is requested and sent to a user's computer.

Web pages are the main feature of the world wide web, but there are several other features that you need to know about. These are listed in Table 3.1.

KEY TERMS

hypertext: a type of text displayed on a screen that contains links to other text

hypertext markup language (HTML): a type of language used to build web pages

TABLE 3.1: FEATURES OF THE WORLD WIDE WEB

Feature	Description
Hypertext information system	**Hypertext** displayed on a web page is a link to another web page, document or some other kind of media (e.g. a video). Hypertext is one of the key underlying features of the world wide web, and many web pages are written in a language called **hypertext markup language (HTML)**. The hypertext information system feature of the world wide web makes it easier for users to navigate around it.
Cross platform	The world wide web can be described as cross platform. This means that it is possible to use it on many different types of device, such as computers, laptops, smartphones, tablets, smart TVs and games consoles.
Distributed	The world wide web can be described as distributed as it is spread across many different pieces of hardware in different locations, rather than using a single piece of hardware in one location. This is beneficial as it spreads out the network traffic.
Open standards	Standards are guidelines that technologies must follow so that they are able to communicate effectively with each other. An open standard is open to use by everyone. The Internet Society is an organisation that helps manage the open standards used by the world wide web.
Dynamic	The world wide web can be described as dynamic. This is because the content that makes up the web is regularly changed and updated.
Web 2.0	Web 1.0 was the first main version of the world wide web. This version of the web heavily depended on a small number of people creating content for a much larger audience to simply read. This is why it became known as the 'read-only' web. Web 2.0 is the second version of the web that developed. This version of the web evolved into a much larger number of users participating and contributing content for the web. This is why it became known as the 'participative social web'.

SKILLS

CREATIVITY

ACTIVITY 2

Use the internet to find out about web 3.0.

What is different about web 3.0, and how will it affect users, software developers and web designers?

KEY TERMS

static web page: a web page in which the content is the same for every user that views it (unless changed by the author)

web authoring: a process of writing the content that is needed to build a web page

STATIC AND DYNAMIC WEB PAGES

The word 'static' means that something stays the same, and the word 'dynamic' means that something can change. Web pages can be static (where the content remains the same) or dynamic (where the web page is regenerated every time it is requested, which means it can change).

A **static web page** is normally written in **web authoring** languages such as HTML, CSS and JavaScript. When a web server receives a request from a web browser for a static web page, it has little processing to do other than sending the data for the web page back to the web browser. The static web page will display the same content for every user that requests the web page. A static web page is used when a website has a small number of web pages that do not regularly need updating. The benefits of a static web page are the following.

KEY TERMS

cache: a type of storage that stores the most frequently requested web pages and data

client-side scripting: processing for a web page that takes place on the client's computer

dynamic web page: a web page where the content is generated every time the page is requested, so it may change and display differently for each user

server-side scripting: processing for a web page that takes place on the web server

- It is easy to build.
- It costs less to create.
- It can be faster to load.
- It can be easier to **cache**.

A **dynamic web page** normally incorporates web authoring languages such as PHP, ASP and AJAX. When a web server receives a request from a browser for a dynamic web page, it may have some additional processing to do before it sends the data back for the web page. This processing may involve searching a database for data about the user to allow data on the web page to be personalised for the user. This could include changing the language used for the web page based on the user's location. The dynamic web page may display different content for every user that requests the web page. A dynamic web page is used when a website has a large number of web pages that regularly need updating. The benefits of a dynamic web page are the following.

- It can be personalised to each user, which can improve user experience.
- It is easier to make site-wide updates.
- It is more scalable, as the server can create a web page when it is needed. The data is stored in a database, and the server can extract whichever data is needed to make the page.

For web pages to be displayed, a practice called scripting needs to be used. Scripting can either be **client-side scripting** or **server-side scripting**. The main difference between the two is:

Client-side scripting takes place on the client computer, normally the user's computer.

Server-side scripting takes place on the web server.

Table 3.2 lists some other differences between client-side and server-side scripting.

TABLE 3.2: CLIENT-SIDE SCRIPTING VS SERVER-SIDE SCRIPTING

Client-side scripting	Server-side scripting
Normally used by static web pages	Normally used by dynamic web pages
It is run by the web browser	It is run by the server
Does not require any interaction with the server	Requires interaction with the server
Reduces the amount of processing that needs to be done by the server	Increases the amount of processing that needs to be done by the server
Main scripting language is JavaScript	Main scripting language is PHP
A user can view the source code that is used for client-side scripting	A user cannot view the source code that is used for server-side scripting

SKILLS
ANALYSIS

ACTIVITY 3

Use the internet to find two websites that use client-side scripting and two websites that use server-side scripting.

- What is the difference in user experience between them?
- What are the factors that have influenced the web developers to use their chosen scripting type?

OPERATING ONLINE

ONLINE WORKING ENVIRONMENTS

KEY TERMS

online working environment: a place where people share resources and communicate with each other about their work, using the internet

virtually: interacting using the internet instead of interacting in a physical place

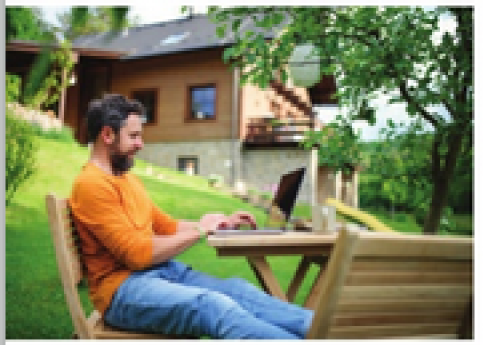
The development of the internet and the world wide web has meant that **online working environments** can be created. This has changed the way in which many people work.

An online working environment is one that exists **virtually** instead of people working face to face in an office. It allows workers to share and work together on resources in a similar way to how they might work if they were face to face in an office. Some well-known examples of online working environments are:

- Adobe Connect
- Google Workspace
- Microsoft SharePoint
- Workplace from Meta.

Advantages and disadvantages of online working environments

Online working environments have given individuals and organisations more flexibility in their working practices. They have had a big impact for individuals and organisations, with the following benefits.


▲ Online working environments mean that there is greater capacity for remote working

Greater capacity for remote working – individuals have more choice over where they want to work. Many organisations have hybrid working arrangements. This means that individuals can choose to work in an office with others or work from home using an online working environment. This allows the individual to choose where they feel happiest working and where they will be most productive. It can also reduce the need for commuting and travelling.

Organisations can reduce their costs – organisations may be able to have smaller office spaces as not all workers need to be in the office at the same time. This means that they can rent smaller office spaces with lower running costs, such as reduced electricity costs.

Greater ability for collaboration – it is much easier for individuals to collaborate wherever they may be working from. Workers can work on shared documents and resources that they can all contribute to very easily. This reduces the need for individuals to send documents to each other using email and avoids multiple versions of a document.

Easier to organise meetings or training – many online working environments have dedicated communication tools such as video conferencing tools. These can make it much easier for individuals and organisations to have a meeting. Many also allow people to use sharing facilities such as sharing their screen, so that others can see a document that they are editing.

There are also some disadvantages of online working environments. For example, consider the following.

Lack of security – remote workers may not have the same robust security measures in place on their home network and computer. One user's compromised computer or network could have an effect on other users connected to the online environment. For this reason, online working environments need to be very secure and have strict access controls in place.

Weaker data protection – data that is available in the online working environment can easily be intercepted as it passes over the internet. Very secure login and encryption protocols need to be used.

▲ Figure 3.3 An online working environment is like a virtual office

SKILLS
CREATIVITY, COLLABORATION

ACTIVITY 4

Use the internet to find out about the facilities that are available in three different online work environments. Using this information, write a design brief for your own online work environment. Make a short presentation about your online work environment.

Communicate with other class members and form a group of four. Each person should show their presentation about their online work environment and the group should vote for the one they think is the best.

You could then show the best ones voted by each group to the whole class, and the class could vote on the best one overall.

DIGITAL FOOTPRINT

KEY TERM

digital footprint: a record of all of a person's activity online

When you walk on any soft ground, like mud or snow, you leave footprints. These footprints are a trail that shows where you have been and which way you have walked. It may also be possible to tell who left the footprints by matching the shape and size of the footprint to the shoe that left it. This kind of footprint is your physical footprint. Your **digital footprint** is similar. It is a record of all the things that you do online, including all the places you access online, making virtual footprints. Your digital footprint includes:

- all the websites that you visit
- all the information that you read online, such as news articles and blogs
- all the emails and messages that you send and receive

- all the data that you enter online
- all the files that you download
- all your social media posts
- all the ways that you interact with social media, such as liking a post.

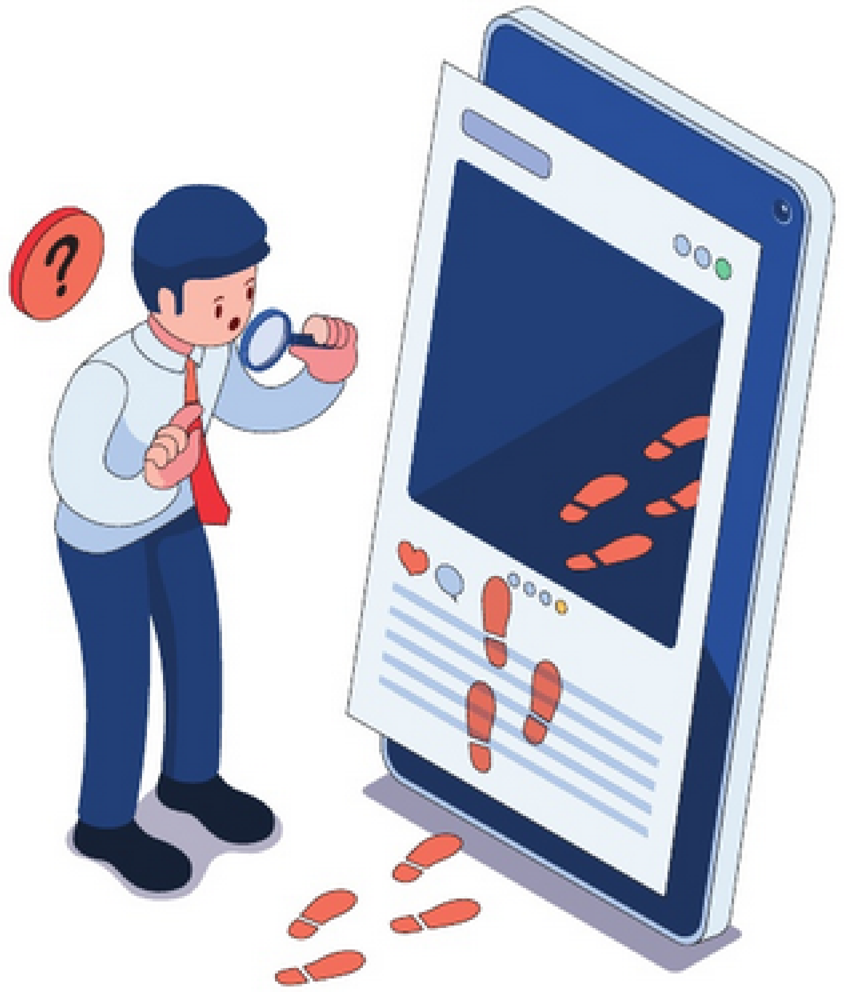


Figure 3.4 Your digital footprint is a record of all your online activity

All these things show a trail of where you have been online and what data you have entered and viewed. You will have both an **active digital footprint** and a **passive digital footprint**.

KEY TERMS

active digital footprint: data you share online deliberately

passive digital footprint: data that is collected about your activity online that you are not aware of

Active digital footprint

This is where you have deliberately shared data online. This can be through sending an email, buying a product or posting on a social media platform.

Passive digital footprint

This is where data is collected about you without you being aware of it. This is data such as a website recording, how many times you have visited it and where you are located.

Why does your digital footprint matter? It is important because it is a permanent record of your activity online. Once this data is collected you have little control over how it is used. The record of your activity online can be a reflection of your reputation. This means that any future employer can look for your digital footprint and find out some of your past online activity. This may affect how they feel about employing you. Criminals can also take advantage of your digital footprint and use it in an attempt to steal your identity and commit fraud.

SKILLS

ANALYSIS, CRITICAL THINKING

ACTIVITY 5

Create a table about the positive and negative effects of a digital footprint. Include both active and passive types of digital footprint in your table.

SKILLS

CREATIVITY, CRITICAL THINKING

ACTIVITY 6

Write down five things that you can do to make sure that you are happy with the digital footprint that you leave. Discuss your answers with a partner.

ONLINE COMMUNITIES

An **online community** is a place where people communicate and interact, which is done using the internet. Online communities are normally created by an organisation, or by an individual creating an online group where people can share and interact. An online community normally exists for people who have a shared interest. These interests could be personal (e.g. TV shows, hobbies, video games) or professional (e.g. software development, professional profiles).

KEY TERMS

monetise: earn money from

online community: a place that exists online where people interact with each other by having discussions and sharing resources

Online communities normally exist on online community platforms. Some common online community platforms are:

- Facebook – a social media site for a person to share information with family and friends about events in their life
- Instagram – a place for a person to show snapshot images of their daily life
- Reddit – to interact with multiple communities about a wide range of interests
- TikTok – to share short videos that can be informative or for entertainment value
- Pinterest – to gather ideas about shared interests or personal projects
- WeChat – a free messaging and calling app.

Why do people join online communities? Mainly because we all like to feel that we belong. There can be several reasons why a person may join an online community, including:

- belonging – we like to feel connected, and many of us find this connection in online communities. This can often help people to feel less lonely.
- shared purpose – we often feel like we have our most engaging and interesting conversations when we speak to someone who has a shared interest.
- increasing knowledge – we can learn more about a shared interest or profession from those who share our interest or profession. Group members will share content that can increase the level of knowledge that we have, and we can also teach others with what we know.
- global connection – we can connect and share our interests with people from all around the world.

Some individuals and organisations began to see the business opportunities that could be possible with online communities and began to **monetise** them. Table 3.3 shows several ways that online communities have been monetised.

TABLE 3.3: MONETISING ONLINE COMMUNITIES

Monetisation method	Description
Targeted advertising	It is possible for those who own the online community platform to allow advertisements to appear within the online community space. If these adverts are targeted, they can be aimed at the common interest that the online community shares. For example, in an online community for video game players there may be targeted advertising about new video games and new games consoles. The owner of the online community can charge advertisers a fee to allow this type of advertising to appear in their online community.

(Continued)

TABLE 3.3: MONETISING ONLINE COMMUNITIES (*Continued*)

Monetisation method	Description
Pay-per-click advertising	In this type of advertising, the owner of the online community allows a company to have an advert displayed in the online community space. If a member of the community clicks on the advert, the company will pay a fee to the owner of the community for each click.
Selling customer data	An owner of an online community platform will often require a person to agree to terms and conditions when joining the online community. One of the terms and conditions could be to allow the owner of the community to sell their data. A company, for example an IT supplier, may choose to buy the data for the members of an online community about computing. They could then use this data to send email adverts to the members about new IT products.
Paid subscriptions	An owner of an online community platform may decide to have part of the community behind a **paywall**. This means that members will need to pay a subscription fee to access all the content available in the community; otherwise they will only be allowed to access limited content. For example, in an online community about fitness and nutrition, members may be able to see a limited number of videos and recipes that they can use. If they pay the subscription fee, they will be able to see a much wider range of videos and recipes, have access to meal plans and be able to access a subscription member-only discussion area.
Sponsored content	An owner of an online community platform may have a company that sponsors them. If you have listened to a podcast, you may have heard at the start of the podcast that today's episode is sponsored by a certain company. The company will pay a fee to the owner as a sponsorship fee. This is another form of advertising for the company.

KEY TERM

paywall: a restriction placed on content that requires users to pay a fee to view the content

▶ Figure 3.5 Pay-per-click advertising is one way that an online community can be monetised

EXAM TIP

When you are given discussion-level questions about online working environments and online communities, you will often be given their use within a context. Make sure that all the points that you make in your answer are related back to the context and are not just left as generic points about online working environments and online communities.

SHARING PERSONAL DATA ONLINE

You have probably shared some part of your personal data online at some point. This may have been as part of an online community. Even simply sharing your name on a social media site is an example of sharing your personal data. You can often discover a lot of information about a person through simply knowing their full name. It is relatively difficult to use any online resources without sharing your personal details. Many organisations require you to disclose them to sign up for an account or service.

Security risks

When you share your personal data online it will be stored by an organisation. There are several security risks for your personal data. The data could be:

- viewed by any third party, especially if you have few privacy settings in place on accounts such as social media accounts. This means that a third party could gather lots of different information about you and create a profile of you. They could use this to steal your identity and commit fraud.
- hacked by a third party and stolen. This could result in your online accounts being accessed and used for malicious purposes, such as to commit a crime.
- stored without correct security methods, which could lead to a data breach or data loss if it is accessed by an unauthorised person.
- obtained from you by dishonest means. A third party could send you an email to make you think that they are a legitimate company that you need to give your personal details to. This is called **phishing**.

KEY TERM

phishing: a scam where a legitimate-looking email is sent to a person to try and obtain their personal data

SKILLS

CRITICAL THINKING

ACTIVITY 7

Write down your answers to these questions:

- What counts as personal data?
- Do you know how many times you have shared your personal data online?
- Do you know which online organisations store your personal data online?

Discuss your answers with a partner. Did they write down or include anything that you did not?

Protecting your data

As your personal data is so valuable, it is very important to know what you can do to help keep it safe and how to be cautious when sharing it online. In your everyday life you probably have several measures that you put in place each day to help keep you and the things you own safe. You may:

- have a lock on your door
- avoid talking to strangers
- keep to routes and areas that you know are the safer places.

There are similar measures that you can put in place to help keep your personal data safe online. Table 3.4 shows some of these.

KEY TERMS

encrypted: data that has been scrambled so it is meaningless to anyone who does not have the key to unscramble it

strong password: a password that has a mixture of letters, numbers and symbols

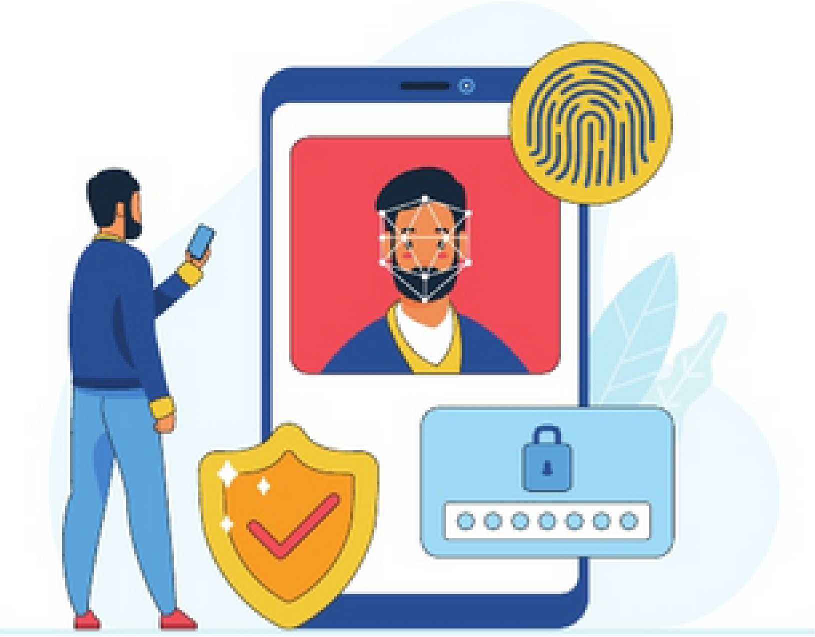
virtual private network (VPN): a secure connection between devices that uses encryption to send data

KEY TERM

hacking: unauthorised access to data

TABLE 3.4: SECURITY MEASURES TO KEEP DATA SAFE

Security measure	Description
Strong password	You probably have a password set for one or more of your devices. It is important to make sure that the passwords that you set are strong passwords. This means that they must: • be long (at least eight characters) • have a mixture of letters, numbers and symbols • have both uppercase and lowercase letters • not contain any personal details or recognisable words. If a password is a strong password, it is much more difficult for a hacker to guess. Many successful hacks have happened because a user has a simple password, e.g. password1234, or includes personal details, e.g. their pet's name.
Biometric device	A biometric device allows you to use a biometric password. This means that instead of using a text-based password you can use your own biological data as a password. Some biometric features that can be connected to or built into a device are: • a fingerprint scanner • a facial recognition scanner • an iris recognition scanner • voice recognition software. It is possible for many people to have the same text-based password. It is also possible for a hacker to guess or work out a text-based password. However, a biometric password, such as your fingerprint or your iris, is unique to you. It is also part of you. This means that it cannot be stolen in the same way that a text-based password can be stolen, and it cannot be guessed because it is your biological data. Therefore, it is much harder to hack a device that is accessed using a biometric password.
Encryption	Any personal data that is stored by you or by an organisation should be **encrypted**. This means that an encryption key (a type of algorithm) is used to scramble the data and make it meaningless to anyone who tries to steal it. The only way to decrypt the data is with the correct decryption key. This helps keep the data safe when it is stored and when it is being transmitted from one device to another. One way to see if an online organisation encrypts any data that is stored and sent by their web server is to see if the URL begins with https. The 's' means secure, which means that data sent and stored by the website is encrypted. Another specific security measure that can be used is a **virtual private network (VPN)**. This creates an encrypted connection for all your communications.
Firewall	A firewall can be hardware based or software based. It examines traffic coming into and out of your device. It can help stop malicious traffic (e.g. malware) entering your device and damaging your data. It can also help stop a third party **hacking** your device and stealing your personal data.
Thoroughly checking emails and URLs	You should check any link that you click when online to make sure that the URL that is attached to it is legitimate. You can do this by making sure the URL is spelt correctly and is the correct URL for the website that you want to access. You should also check any information that is sent to you in an email asking for your personal data. You should make sure that the email is from a legitimate company by checking if the email address it has been sent from is correct and that everything in the email is correctly spelt.
Privacy settings	It is important when creating any online account, for example for a social media platform, that you make sure you have the most secure level of privacy settings in place. This will reduce the risk of a third party looking at your account and being able to access lots of your personal data. You should also be very cautious about what personal data you share on your social media, as a third party could monitor your profile and find out a lot of information about you by piecing together small bits of data. For example, you may regularly take a picture outside your local coffee shop with your daily coffee. If you post this several times, a third party can begin to gain an understanding of where you live. It is not likely that you would tell a stranger on the street your address, so be as cautious as this with any data that you share online.



▶ Figure 3.6 Using a biometric device can help keep your personal data safe

SKILLS

CREATIVITY, CRITICAL THINKING

ACTIVITY 8

Choose three security measures that can be used to help keep your personal data safe. Create a presentation showing three ways that a person can help keep their personal data safe online.

Share your presentation with a partner.

KEY TERMS

artificial intelligence (AI): an IT system that has the ability to learn and adapt. It is designed to mimic the intelligent behaviour of a human

cloud computing: cloud storage and other services such as software applications that are available from a cloud computing organisation

cloud storage: servers that provide data storage that you can access from a remote location using the internet

THE CLOUD

CLOUD STORAGE

Most of the data that you or an organisation stores online is stored in the cloud. **Cloud storage** is a method of storage where data is stored on hardware (such as servers) and is accessed from a remote location using an internet connection. This is different to storing data on a storage device at your home (such as your computer), which is called storing your data locally. The hardware that is used in cloud storage usually belongs to a third party, which is normally the organisation that is providing the cloud storage.

CLOUD COMPUTING

Cloud computing is cloud storage along with further services that are offered by an organisation. These services include access to software and resources using cloud storage, as well as the data that is stored.

One example of cloud computing is Microsoft365. With this cloud computing service, Microsoft allows a user access to software applications such as Word and Excel. This software runs in the cloud and is accessed using an internet connection. Cloud computing also allows a user to back up all their documents and resources in cloud storage.

Another example of cloud computing is iCloud. This is a cloud computing service that is offered by Apple, and it allows users to store data such as their photos, videos, documents, notes and contacts in cloud storage. Users can then access this data from multiple devices in multiple locations.

There are benefits and drawbacks of cloud computing services to a user or organisation, some of which are listed in Table 3.5.

TABLE 3.5: ADVANTAGES AND DISADVANTAGES OF CLOUD COMPUTING

Advantages	Disadvantages
There is no need to buy additional software to store data. Individuals and organisations are storing an increasing amount of data. Storage devices can be expensive, and storing data in cloud storage means they do not need to buy new storage devices.	A fee may need to be paid for cloud computing services. Some cloud computing organisations may allow a small amount of cloud storage for free, but to store large amounts of data in the cloud you will have to pay a fee. This is often a subscription fee that is paid on a monthly or yearly basis.
There is no requirement to employ technical people to manage any storage hardware. This could save money for an organisation that may need to employ this kind of staff.	Users have to put their trust in a cloud computing organisation and rely on it to make sure their data is kept safe. If the cloud computing organisation does not have enough security measures in place, the data stored in their cloud storage could be hacked. Each country also has their own privacy laws. A cloud computing company could store your data anywhere in the world. You probably won't know where your data is being stored. The country it is stored in could have privacy laws that are different from your own. For example, they could allow the company to sell your data, allow **artificial intelligence (AI)** to have access to your data or allow different law enforcement agencies to have access to your data.
Users can access their data from any device and any location if they can sign in to their account using an internet connection. This can be a very convenient feature for users.	If a user cannot get access to an internet connection, or an internet connection fails, they cannot gain access to their software and data. If an internet connection is weak, it can also affect how well the user is able to use the software and access their data.
Users do not need to worry about creating a backup of their data as this will be regularly created and stored in cloud storage by the cloud computing organisation.	
Users do not need to worry about updating their software as the software is stored in cloud storage and will be updated there by the cloud computing organisation.	

(*Continued*)

TABLE 3.5: ADVANTAGES AND DISADVANTAGES OF CLOUD COMPUTING (*Continued*)

SKILLS

CREATIVITY, CRITICAL THINKING

ACTIVITY 9

Use the internet to find out what is meant by data synchronisation and how it is relevant to cloud computing.

Create a slide using boxes and lines to explain data synchronisation in cloud computing environments.

THINKING LIKE AN IT SPECIALIST

The digital footprint of people is ever increasing. This is because people use the internet for more and more of their daily tasks and lives. Many people are unaware how much data is stored about their online activity and what kind of information someone can gain access to. What would you do to raise awareness about a person's digital footprint?

CHECKPOINT

1. What is a web server?
2. Explain the role of a DNS.
3. What are three features of the world wide web?
4. Explain the difference between a static web page and a dynamic web page.
5. Give two reasons why an individual would use an online working environment.
6. What is the difference between targeted advertising and pay-per-click advertising in an online community?
7. List three security risks that a person could be subjected to when sharing their personal data online.
8. What is cloud storage?
9. Define a digital footprint.

SKILLS

ANALYSIS, INTERPRETATION, PROBLEM-SOLVING

EXAM PRACTICE

1. People regularly use the internet and the world wide web to access web pages.

(a) Describe the difference between the internet and the world wide web. [2 marks]

(b) Describe two features of the world wide web. [4 marks]

2. Jack types a URL into his web browser to access a web page.

(a) Explain how the URL is converted into an IP address to send a request for the web page. [3 marks]

(b) The web page that Jack views is displayed for him with personalised content. Scripting is used to process the web page but Jack is not able to see the code used for this scripting.

(i) What kind of web page is Jack viewing? Write the correct letter in your exercise book. [1 mark]

A Active web page
B Dynamic web page
C Standing web page
D Static web page

(ii) State which type of scripting is used for the web page. [1 mark]

3. Explain two reasons an organisation would want their employees to use an online working environment. [4 marks]

4. Rosa is the owner of an online community that she monetises.

(a) Describe what is meant by an online community. [2 marks]

(b) Explain two ways that Rosa could monetise the online community. [4 marks]

5. Maryam is concerned about her digital footprint.

Explain two ways that Maryam could make sure that her digital footprint is beneficial. [4 marks]

TOPIC 4
IT SYSTEMS

We use IT systems nearly every day of our lives. They aid us in so many of our daily tasks. In this topic you will see how IT systems can be broken down into smaller parts so that they can be analysed for functions and processes. You will learn how the flow of data can be mapped through an IT system, and how to use a dataflow diagram to do this. You will see how the sequence of processes carried out by an IT system can be represented as a flowchart, and how to evaluate an IT system for a given purpose for an individual or an organisation.

TOPIC 4 IT SYSTEMS

LEARNING OBJECTIVES

By the end of this topic you should be able to understand:

- the main components of an IT system
- how to break a system down into smaller parts
- what dataflow diagrams are used for and how to create them
- what flowcharts are used for and how to create them
- how to design and evaluate part of an IT system and a whole IT system
- the advantages and disadvantages of a given IT system for an individual and an organisation, including the use of digital, peripheral and storage devices.

GETTING STARTED

Make a list of all the IT systems that you see or use in a day. You probably see and use more than you think, such as traffic lights, air-conditioning systems and supermarket self-checkouts.

Find two other people to work with so you have a group of three. Compare your lists and see if the other people in your group have any systems that you may have forgotten to list.

▶ Supermarket checkouts are just one example of an IT system that we might use in everyday life

SYSTEMS DESIGN

MAIN COMPONENTS OF AN IT SYSTEM

KEY TERM

IT system: all the hardware and software that are combined to create a whole system

An **IT system** is more than just a single computer. It is all the different components that work together to create the whole system. The main four components of an IT system are:

- hardware – all the physical devices that are part of the system
- software – all the computer programs that are part of the system
- processes – all the different steps and actions that are carried out using the system
- people – all the users and individuals that use the system.

An example of an IT system is a check-in system at an airport. Table 4.1 shows the four components of this IT system and a description of what the check-in system may have for each component.

TABLE 4.1: FOUR COMPONENTS OF AN IT CHECK-IN SYSTEM

Component	Description
Hardware	There may be several pieces of hardware involved in the check-in system. This could include: • an electronic scale to weigh luggage • a device to scan a person's boarding pass to see which flight they are taking • a computer to process the data of all the customers who are checking in • a keyboard or a touchscreen to enter the data for any boarding passes that may not scan • a display screen to allow a person to see their check-in data • a speaker to play any notification sounds, such as a beep to show that a boarding pass has scanned • a printer to print out any labels that are needed to attach to a person's luggage.
Software	There may be several pieces of software involved in the check-in system. This could include: • a user interface to allow check-in staff and passengers to interact with the system • an operating system to run the basic functions of the check-in system • a database to store all the data of the people that need to check-in • utility software to maintain the check-in system and make sure that it keeps functioning properly.
Processes	There will likely be many different processes involved in the check-in system. These could include: • putting luggage on the weighing scale • scanning a boarding pass • entering a person's data in the system • storing a person's data in the database • retrieving a person's data from the database • waiting for a luggage label to print.
People	The people who may be involved in the system include any person who may interact with the system. This could include: • a person checking in for a flight • an employee of the travel company that uses and helps with the system.

SKILLS

ANALYSIS, COLLABORATION

ACTIVITY 1

Think about an IT system that is familiar to you. You could choose one of your own but if you cannot think of any, here are a few ideas:

- school library IT system
- elevator in a building
- supermarket self-checkout.

Break down the IT system into the four component parts as in Table 4.1.

Show your table with the four components to a friend. Do they agree with what you have noted? Would they change anything about how you have broken down the system? Would they add anything to any of your components?

KEY TERMS

decomposition: breaking down a problem, design or system into smaller parts

sub-system: a smaller individual section of a larger system

DECOMPOSING A SYSTEM INTO SUB-SYSTEMS

Some IT systems can be large and it is helpful to be able to break them down into smaller parts and evaluate them. Breaking an IT system down into smaller parts is called **decomposition** of the system. A smaller part of a larger system is often called a **sub-system**. If we look at the example of the check-in system, this could be broken down into two smaller sub-systems, for example:

sub-system 1 – weighing luggage

sub-system 2 – scanning a boarding pass to check in to the flight.

SKILLS

ANALYSIS

ACTIVITY 2

Think about the IT system that you broke down into the four components in Activity 1. Decompose the system that you chose into as many sub-systems as you can think of.

Share your sub-systems with a partner. Would they have chosen the same ones?

DATAFLOW

REPRESENTING IT SYSTEMS AS A DIAGRAM

It can be difficult to understand all the actions in processes of a large system, including where and how all the data is be used and transmitted around the system. However, it is important for an organisation to know this information. There are two types of diagrams that can be used to improve our understanding of an IT system. These are:

- **dataflow diagrams**
- **flowcharts**.

KEY TERMS

dataflow diagram: a diagram that shows how data flows through different parts of a system

flowchart: a diagram that shows the sequence of a process and decisions that occur in a system

DATAFLOW DIAGRAMS

A dataflow diagram visually illustrates how data moves through a system. This involves showing:

how the data is entered into the system (input)

what processes will be carried out with the data

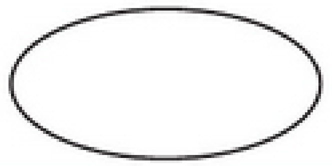
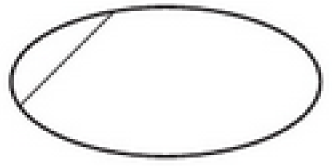
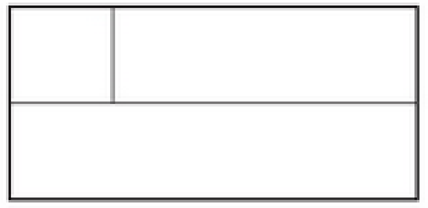
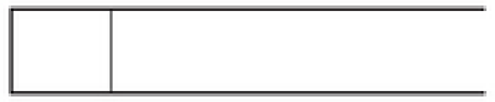
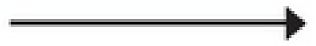
where it is being transmitted to and from

where it is being stored within the system

how the data leaves the system (output).

Table 4.2 contains a set of symbols that you will use to create a dataflow diagram.

TABLE 4.2: DATAFLOW DIAGRAM SYMBOLS AND THEIR PURPOSE

Symbol	Purpose
	This symbol is used to show a data source. This is a place where data is input into the system or output from the system.
	This symbol is used to show a duplicate data source in the system. This is an input into the system or output from the system that already exists somewhere else in the system. An example would be where the customer requests a quotation from an order management system, and later raises an order. The data source is still the customer, but they are interacting at a different point and with different data.
	This symbol is used to show a process that happens in the system.
	This symbol is used to show a data store in the system. This is a place where data is stored within the system.
	This symbol is used to show which way data is flowing through the system; it will often have a label to show what data is being transmitted between two parts of the system.

To create a dataflow diagram for a system, you need to break the system down into the different components that are shown in a dataflow diagram. We can try creating a dataflow diagram for the airport check-in system we looked at earlier in the topic.

We need to break the system down into the following parts:

data sources

processes

data stores.

The data sources could be:

a barcode scanner

weighing scales

a customer.

The processes could be:

scanning the barcode on the boarding pass

- weighing the luggage
- printing a luggage label for the customer.

The data stores could be:

- a database of all customers who have checked in for their flight
- a database of all the luggage that is checked in for each customer for a flight.

The dataflow diagram for this system could look something like this.

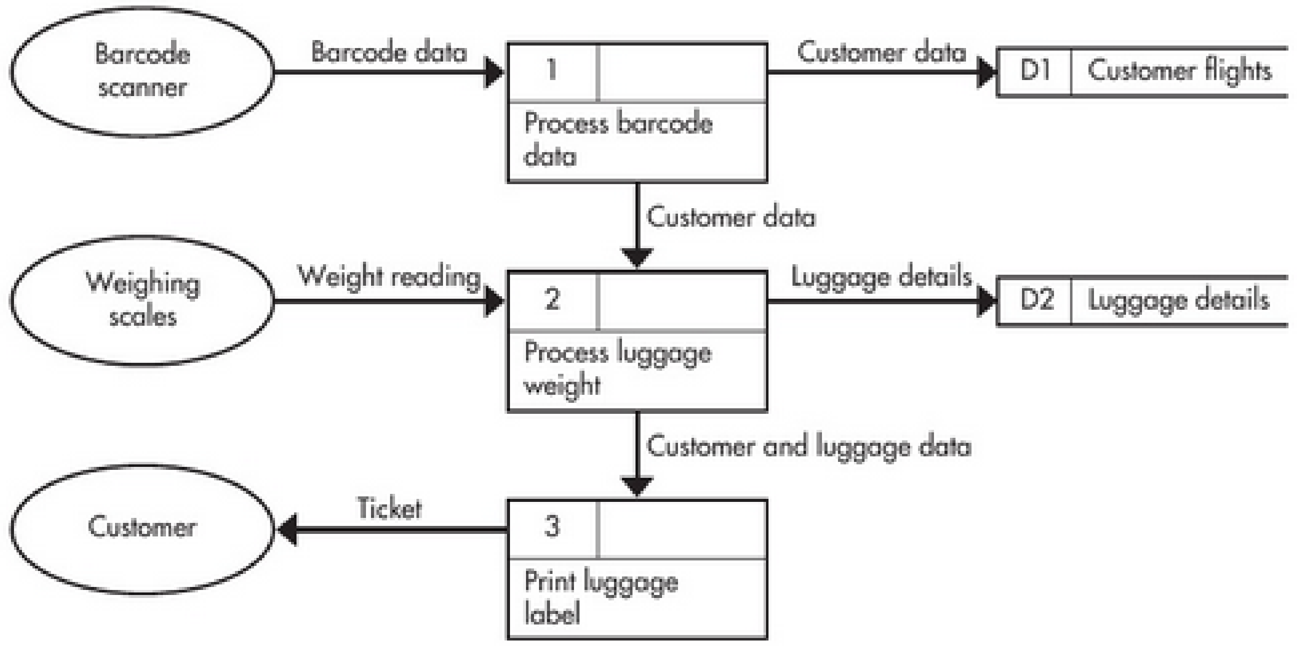


Figure 4.1 Dataflow diagram for an airport check-in system

SKILLS
COLLABORATION, COMMUNICATION

ACTIVITY 3

Work with a partner. Look at the dataflow diagram. Discuss whether you think it is easier to understand how the system works from looking at the lists of what the system involves or the dataflow diagram.

Discuss why you think this. What do you think is clearer about your choice?

Now that you have seen an example of a dataflow diagram, you are going to create one yourself. Remember to look at Table 4.2 to find the correct symbols to use. Make sure that you include data sources, processes and data stores in your diagram.

SKILLS
ANALYSIS, PROBLEM SOLVING

ACTIVITY 4

A toy shop has an online version of their shop where customers can order toys that they want delivered. The online shop has:

- a website where customers can place an order
- a database of all their stock
- a wholesale supplier that they reorder all their toys from in order to restock.

Break the system down into the data sources, processes and data stores that you think are needed. Then create a dataflow diagram for the system.

Compare your dataflow diagram to a friend's diagram. Are your diagrams the same? Discuss any differences you may have and make any changes to your diagram that you think are needed.

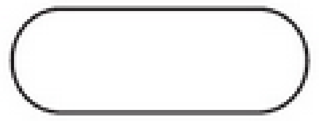
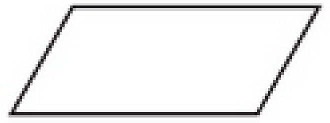
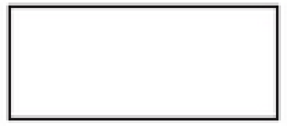
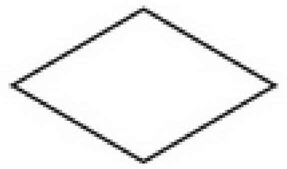
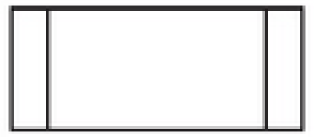
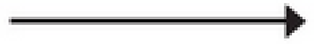
FLOWCHARTS

A flowchart visually represents the steps and decision points in a process within a system. It could include multiple processes within a system. This can involve showing:

- inputs into the system
- outputs from the system
- actions that occur within the process
- decisions that need to be made within the process
- the sequence of the steps that will occur within the process.

Table 4.3 contains a set of symbols that you will use to create a flowchart diagram.

TABLE 4.3: FLOWCHART DIAGRAM SYMBOLS AND THEIR PURPOSE

Symbol	Purpose
	This symbol is called a terminator. It is used to show where a flowchart starts and stops.
	This symbol is used for any input or output of data to the system.
	This symbol is used for any processes that happen within the system.
	This symbol is used for any decisions that need to be taken for the flow of data within the system.
	This symbol is used to show where any subprocesses occur within a system.
	This symbol is used to show the flow of the sequence of actions within the system.

To create a flowchart for a system or process, you need to break it down into the stages that will appear in the flowchart.

Let's look at another example. A teacher sets a music quiz for her class. She has a system where she can input each student's name and score from the quiz and it will output a bronze, silver, gold or participation certificate. The criteria for this will be:

- gold certificate – score of 75 or greater
- silver certificate – score of 65 or greater
- bronze certificate – score of 50 or greater
- participation certificate – score of less than 50.

The system begins by asking how many students she would like to enter. It will repeat the input until all the students have been entered.

We can represent this system as a flowchart by breaking down the part of the system that will be included in the flowchart. For example:

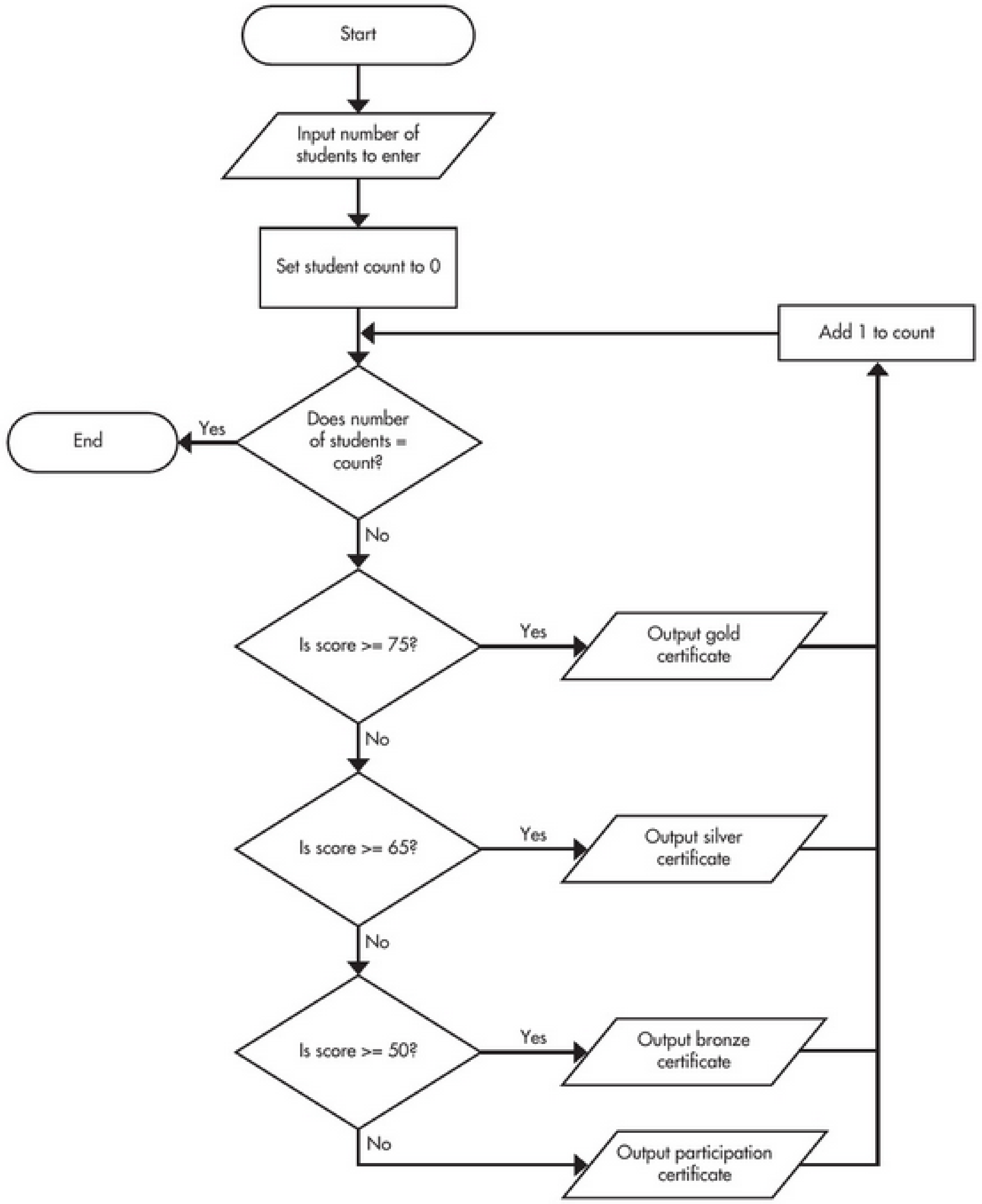
inputs – number of students, student name, student score

outputs – student certificate

processes – store student name, score and certificate type in a database; set count to number of students; add to a count of students entered

decisions – check whether all students have been entered and then compare student score to certificate criteria.

Figure 4.2 shows an example of a flowchart for this system.


Start
Input number of students to enter
Set student count to 0
Add 1 to count
End
Yes
Does number of students = count?
No
Is score >= 75?
Yes
Output gold certificate
No
Is score >= 65?
Yes
Output silver certificate
No
Is score >= 50?
Yes
Output bronze certificate
No
Output participation certificate

▶ Figure 4.2 Flowchart for student music quiz certificate system

SKILLS

ANALYSIS, PROBLEM SOLVING

ACTIVITY 5

A clothing shop has a loyalty card system for customers. Depending on how much they have shopped at the store, a customer may have a diamond loyalty card or a platinum loyalty card.

Customers who have a diamond loyalty card get:

- 10 per cent discount if they spend over $50
- 15 per cent discount if they spend over $100.

Customers who have a platinum loyalty card get:

- 15 per cent discount if they spend over $50
- 20 per cent discount if they spend over $100.

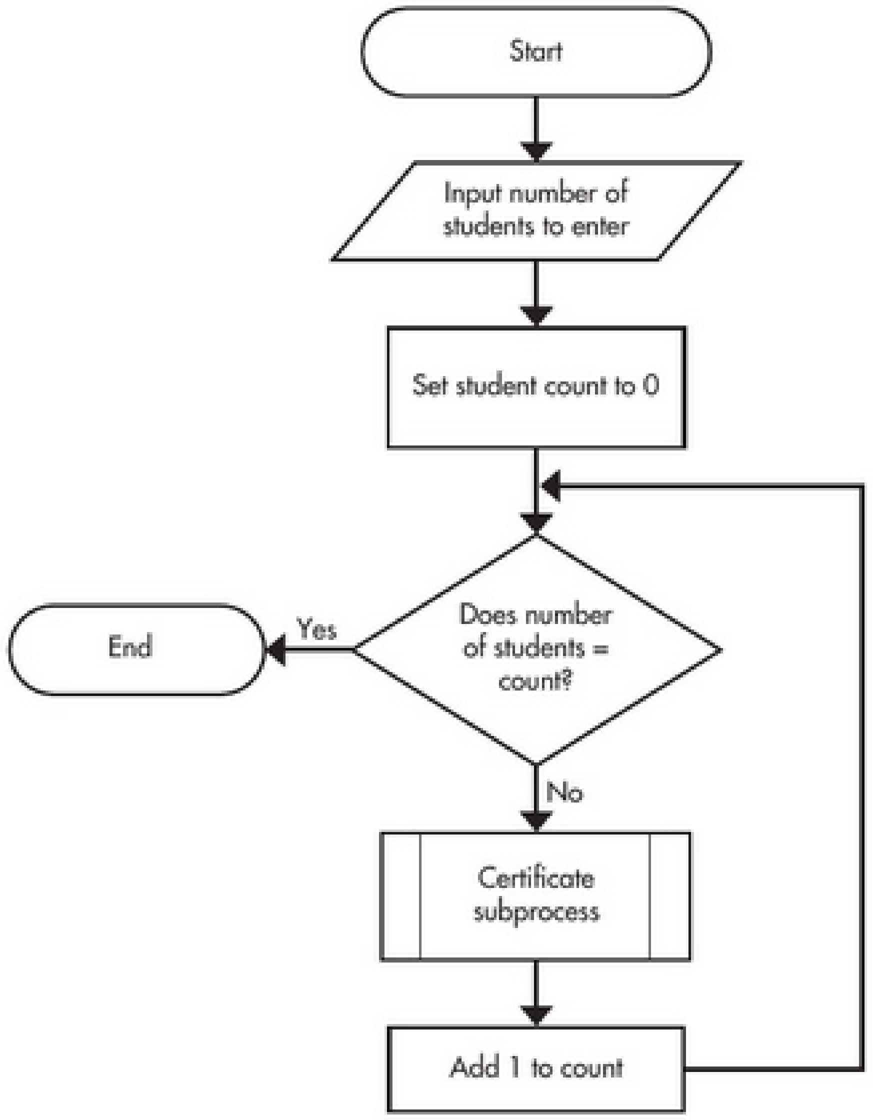
Customers who do not have a loyalty card get a 5 per cent discount if they spend over $50.

Create a flowchart to show the process of calculating and outputting the final cost of a customer's shopping.

Swap your flowchart with a friend. Follow your friend's flowchart through. Can you easily do this? Are there any parts that you are unsure about? Give your friend feedback about any parts that you were unsure about. If you were given feedback about your flowchart, how could you try to change it to improve it?

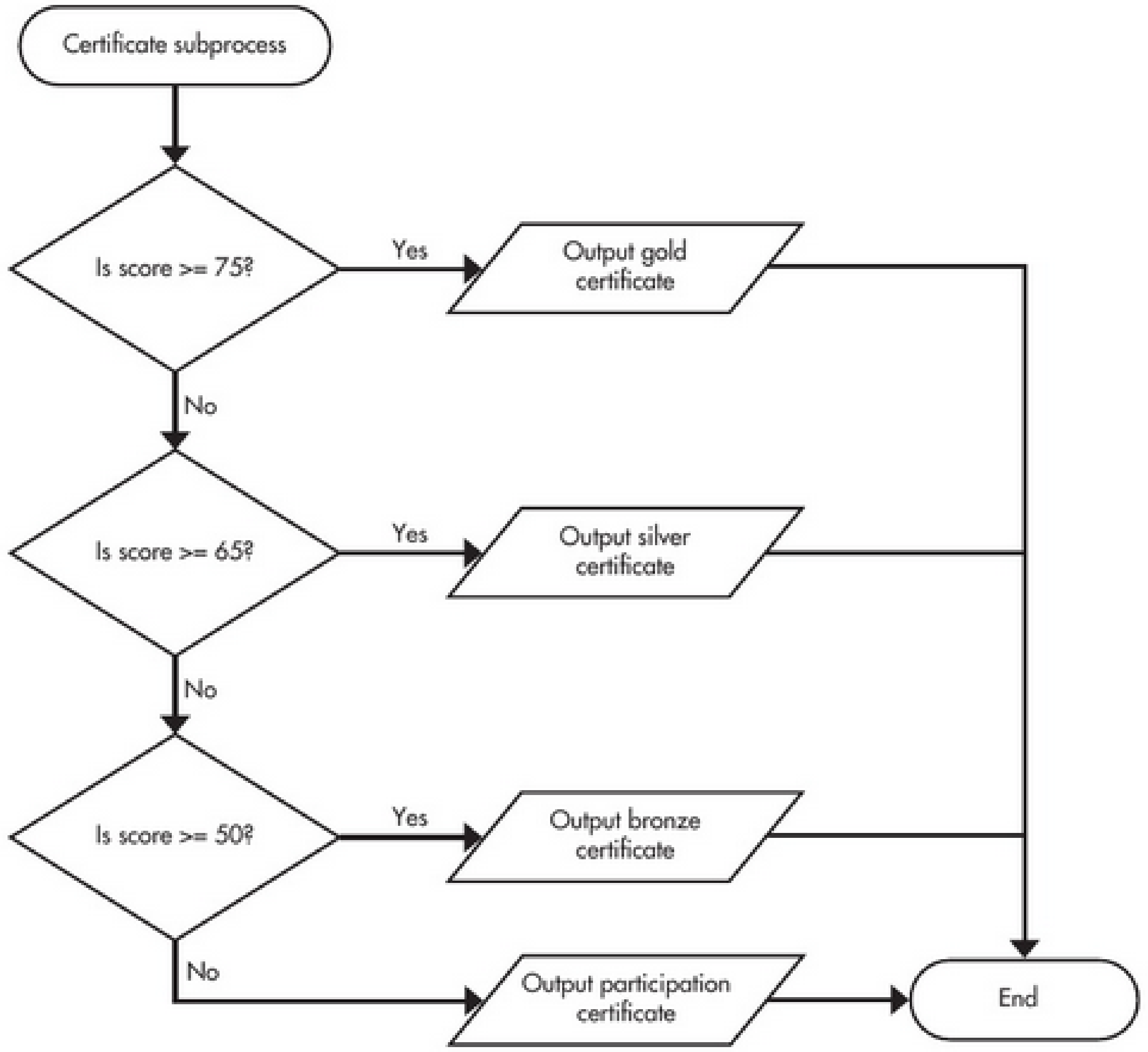
There is one symbol that was not used in the teacher's music quiz flowchart and that is the subprocess. Parts of a flowchart can be split off into a separate section. This is called a subprocess. This is normally done with a flowchart to make the flowchart itself smaller and easier to understand.

We can create a subprocess in the flowchart for the certificate system. The part of the flowchart that decides which certificate a student is awarded can be placed in a separate process. A subprocess symbol will then be used in the flowchart to show where the subprocess will occur. For example, see Figure 4.3.


Start
Input number of students to enter
Set student count to 0
End
Yes
Does number of students = count?
No
Certificate subprocess
Add 1 to count

▶ Figure 4.3 Flowchart for certificate system using subprocess symbol

The subprocess called by using the subprocess symbol in the main flowchart would then be as shown in Figure 4.4.



▶ Figure 4.4 Flowchart for the subprocess of the certificate system

Using a subprocess in this way makes the main flowchart smaller and easier to follow.

EXAM TIP

You should practise breaking down systems into smaller parts and drawing flowcharts and dataflow diagrams for them. You could do this for lots of different kinds of systems that you see and use in daily life, for example a traffic light, a security system, a vending machine and a train ticket machine.

DESIGNING AND EVALUATING AN IT SYSTEM

When you **design** an IT system, you will need to break it down into all the elements that we have spoken about in this topic. You can represent the design of the system using a dataflow diagram and a flowchart. They will both show different aspects of the design of a system. In the exam, you will be asked to draw or complete a diagram for either of them to represent the design of an IT system.

KEY TERM

design: a plan of what the requirements are for a system

▶ Designing an interface for an app

KEY TERMS

design brief: the place where the design for a system is noted

evaluate: to assess and judge how suitable a system is and how closely it matches the design

fit for purpose: an evaluation to see whether a system meets all the original requirements from the design

peripheral: a piece of hardware that is attached to a computer such as a keyboard, mouse, printer and headset

When you **evaluate** an IT system, you will also need to think about all the elements that we have spoken about in this topic. One of the most important things that you will need to consider is a concept called '**fit for purpose**'. This is where the original requirements of a system are assessed against the IT system. If the final IT system meets the original requirements, it can be deemed fit for purpose. This means that it is suitable for the purpose for which it is intended. Simply, it has been built to meet all the requirements that were set out during the development of the system. To evaluate if a system is fit for purpose, there will need to be clear requirements set out in the development of the system. The end system can then be measured against the criteria.

Often the requirements for a system will be tailored to the needs of an individual or organisation and the evaluation will decide whether the needs have been met. One of the reasons an individual or organisation will pay a lot of money for the development of a bespoke IT system is so that it meets their specific needs. Therefore, an evaluation to show whether the system is fit for purpose is crucial.

To evaluate an IT system there are several questions that you may need to consider.

What digital devices are included in the system?

What **peripheral** devices are included in the system?

What storage devices are included in the system?

What processes are carried out by the system?

Does the system meet each of the requirements set out in the **design brief**?

What are the advantages of the IT system?

What are the disadvantages of the IT system?

All of these questions will provide a lot of information that can be used to decide whether a system is fit for purpose.

For example, consider this design brief from a restaurant:

We want an IT system that will allow users to see a menu and order food from their table. We change the menu regularly, so it must be easy to update. The order must be sent through to the kitchen and be easy for all the chefs in the kitchen to see. We want to keep costs to a minimum for the system and do not want to have to buy many new devices. We are nervous about having a system that is based online as it cannot be used if the internet connection is lost.

This is a description of the final system that is created:

'There are paper menus on each table in the restaurant that have a QR code that a customer can scan with their mobile phone. The QR links them to an online booking system that can be used to select items for their order and send the order to the kitchen. There is a computer with a printer in the kitchen that receives the order and prints it out for the chefs to see. Each order is stored in a database on a 1 TB hard disk drive in the computer.'

We can evaluate the system by considering the questions from earlier:

What digital devices are included in the system?

One digital device that is included in the system is a customer's mobile phone.

What peripheral devices are included in the system?

One peripheral device that is included in the system is a printer.

What storage devices are included in the system?

One storage device that is included in the system is the hard disk drive that is in the computer.

What processes are carried out by the system?

One process carried out by the system is the order being sent from a user's mobile phone to the computer in the kitchen.

Does the system meet each of the requirements set out in the design brief?

The system meets the needs of the users as they can easily see the menu and order from their table. The restaurant stated that they were nervous about the system being based online. They didn't state that they did not want it to be online, but they could feel that this requirement has not been met.

What are the advantages of the IT system?

One advantage of the system is that the customer is able to easily order from their table.

What are the disadvantages of the IT system?

One disadvantage of the system is that the order for the kitchen is printed out and this could easily be lost.

SKILLS
CRITICAL THINKING

ACTIVITY 6

Evaluate the system by thinking of at least one more response to each of the questions that you have just read. Ideas that could help your focus are:

- How easy is it to update the menu?
- Will a 1 TB hard drive be sufficient storage space?

THINKING LIKE AN IT SPECIALIST

Flowcharts are a very useful planning tool for visualising the sequence in which things happen in a process. They are regularly used by IT specialists and computer programmers to plan a system. Why do you think there are different symbols for different actions that can be taken in the flowchart? Would it not be simpler to have a rectangular box for each action?

CHECKPOINT

1. What is an IT system?
2. What is meant by decomposition of a system?
3. List the four main components of an IT system.
4. What does a dataflow diagram show?
5. What does a flowchart show?
6. Draw three symbols that are used in a flowchart.
7. Explain what 'fit for purpose' means.
8. What are three questions that you should consider when evaluating an IT system?

SKILLS
INTERPRETATION, PROBLEM SOLVING

EXAM PRACTICE

1. Hardware is one component of an IT system.
   (a) Give one other component of an IT system. [1 mark]
   (b) IT systems can be broken down into sub-systems. State the name given to the process of breaking a system down into sub-systems. [1 mark]
   (c) After an IT system is developed it is evaluated for whether it is 'fit for purpose'. Explain what is meant by 'fit for purpose'. [2 marks]
2. A social media platform allows users to create an account. To create an account, the user needs to enter a username. When the user has entered their username, the system checks whether it is eligible. The requirements for the username are as follows:
   - It needs to be at least six characters in length.
   - It cannot contain any special characters
   - It cannot already exist in the username database.

   If any of the requirements are not met, the user must enter a different username. Once the username requirements are all met, the username is stored in the username database.

   Draw a flowchart to show how the process of setting a username works. [6 marks]
3. A company sells paper and allows customers to order boxes of paper online.

   Customers use the website to place an order online.

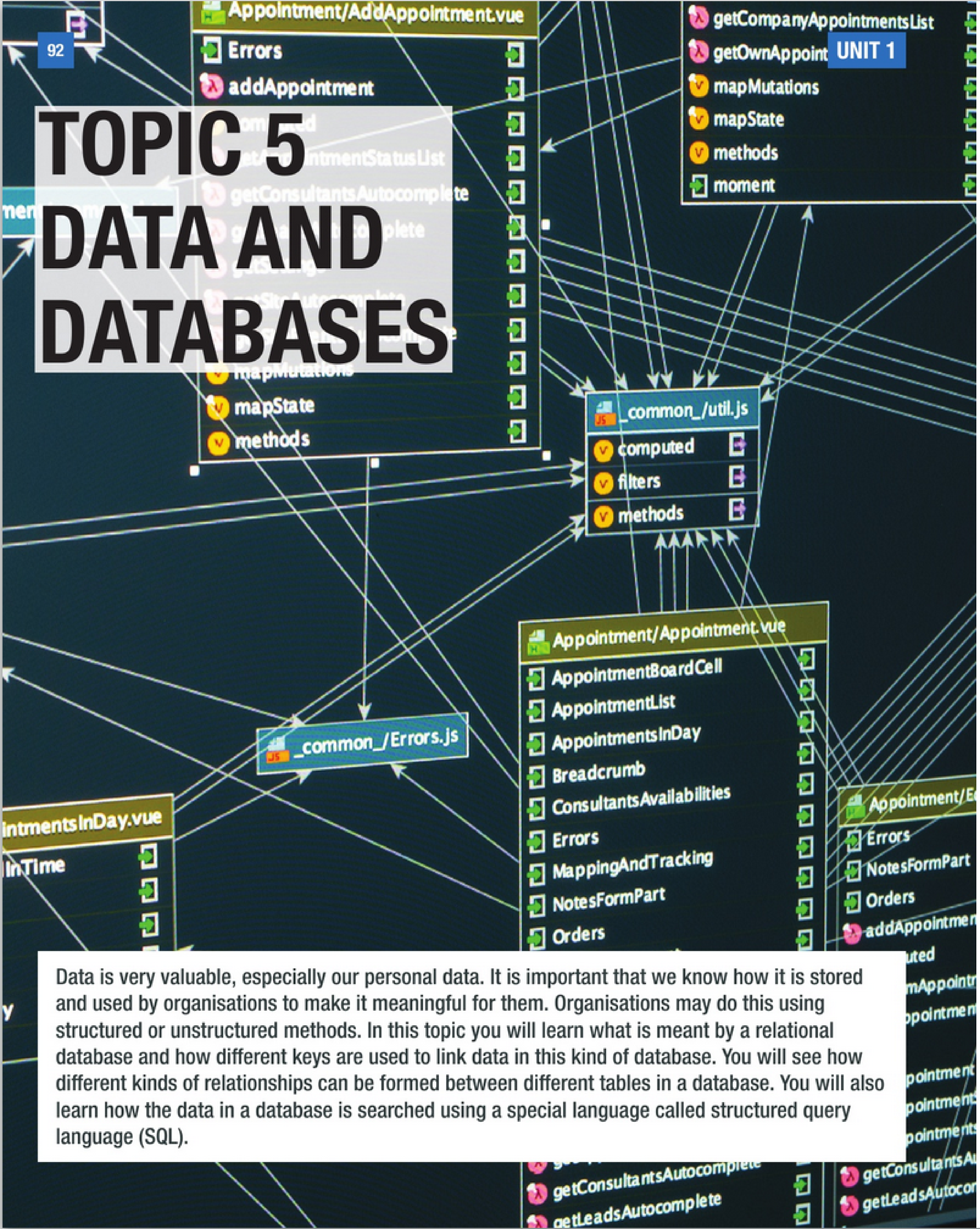
   The order is processed and stored in the order database.

   The stock database is updated to deduct the paper ordered.

   At the end of the day a report is generated about the stock levels.

   If the stock is low for a certain type of paper, a restock order is automatically sent to the wholesaler to order more paper.

   Draw a dataflow diagram to show the flow of data through this process. [6 marks]


Appointment/AddAppointment.vue
Errors
addAppointment
getCompanyAppointmentsList
mapMutations
mapState
methods
moment
mapState
methods
common/util.js
computed
filters
methods
Appointment/Appointment.vue
AppointmentBoardCell
AppointmentList
AppointmentsInDay
Breadcrumb
ConsultantsAvailabilities
Errors
MappingAndTracking
NotesFormPart
Orders
getConsultantsAutocomplete
getLeadsAutocomplete
common/Errors.js
InTime
Errors
NotesFormPart
Orders

TOPIC 5 DATA AND DATABASES

Data is very valuable, especially our personal data. It is important that we know how it is stored and used by organisations to make it meaningful for them. Organisations may do this using structured or unstructured methods. In this topic you will learn what is meant by a relational database and how different keys are used to link data in this kind of database. You will see how different kinds of relationships can be formed between different tables in a database. You will also learn how the data in a database is searched using a special language called structured query language (SQL).

TOPIC 5 DATA AND DATABASES

LEARNING OBJECTIVES

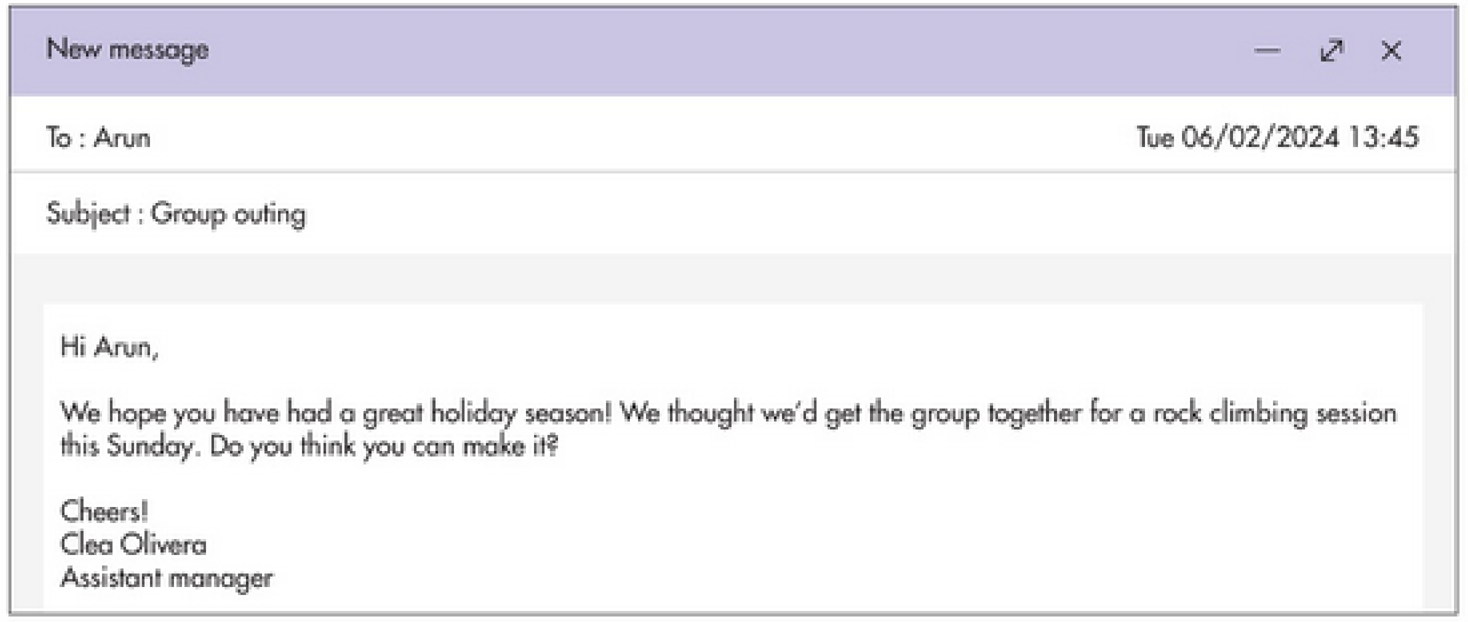
By the end of this topic you should be able to understand:

- what is meant by data and how it is different from information
- what is meant by structured and unstructured data
- the benefits of data extraction to create meaningful data for organisations
- the different components of a database, including relational databases and the different types of keys
- the different types of relationships that can be created between tables in a database
- how to create queries using structured query language (SQL).

GETTING STARTED

You have probably received hundreds of emails in your lifetime. You pay attention to the main contents of the email, but have you ever thought about the different types of data that the email contains?

Look at the following email:



New message

To : Arun

Tue 06/02/2024 13:45

Subject : Group outing

Hi Arun,

We hope you have had a great holiday season! We thought we'd get the group together for a rock climbing session this Sunday. Do you think you can make it?

Cheers!
Clea Olivera
Assistant manager

Think about what you already understand from your everyday life about the words 'structured' and 'unstructured'. What do they mean?

1. Can you identify in the email what is structured data?
2. Can you identify in the email what is unstructured data?

Work with a partner. Share your thoughts about:

- what you think is structured data, and why
- what you think is unstructured data, and why.

Discuss any differences that you have and come to an agreement about what you think is structured and unstructured data. When you have read the section in this topic about structured and unstructured data, see if you were correct!

81%
69%
85%
65%
37%

▶ We can access data easily but how useful is it?

DATA AND INFORMATION

You may think data and information are the same thing, but they're not! Data is text, values, symbols, images or sound that have not been given any meaning. We see them but they do not have any meaning for us.

For example, look at the data:

1008

What does this data mean? It is just a value to us. It could be a year, the year 1008. It could be the number of people at a football match. It could be the price for a product. Without any meaning, we are just guessing.

We can then add meaning to it to make it into **information**.

There are 1008 official Pokémon!

This is now information as the data has been given a **context** that gives it meaning.

Therefore:

Data + Meaning/Context = Information

This means that information is data that has been given a context to give it meaning.

Data can be one of two different structural forms; it can either be structured data or unstructured data.

STRUCTURED DATA VS UNSTRUCTURED DATA

STRUCTURED DATA

Structured data is exactly what it sounds like it would be. It is data that has a structure. This structure is often built around the **data type** of the data. Data is given a structure to make it easily searchable. This structure is normally a set **format** that matches the data type. This is why most of the data that is stored in a database is an example of structured data.

KEY TERMS

context: the conditions in which something exists

data type: a way of grouping data; examples include text, number, currency, date, time and dates

format: the way in which data is arranged

information: data that has been processed by giving it context and meaning

structured data: data that has a specific format

KEY TERM

machine learning: a type of artificial intelligence that uses data and algorithms to learn and adapt its processes

For example, the price of a product is $1.95. This is an example of structured data. The data type it has could be 'currency' or 'float'. It also has a set format of two decimal places.

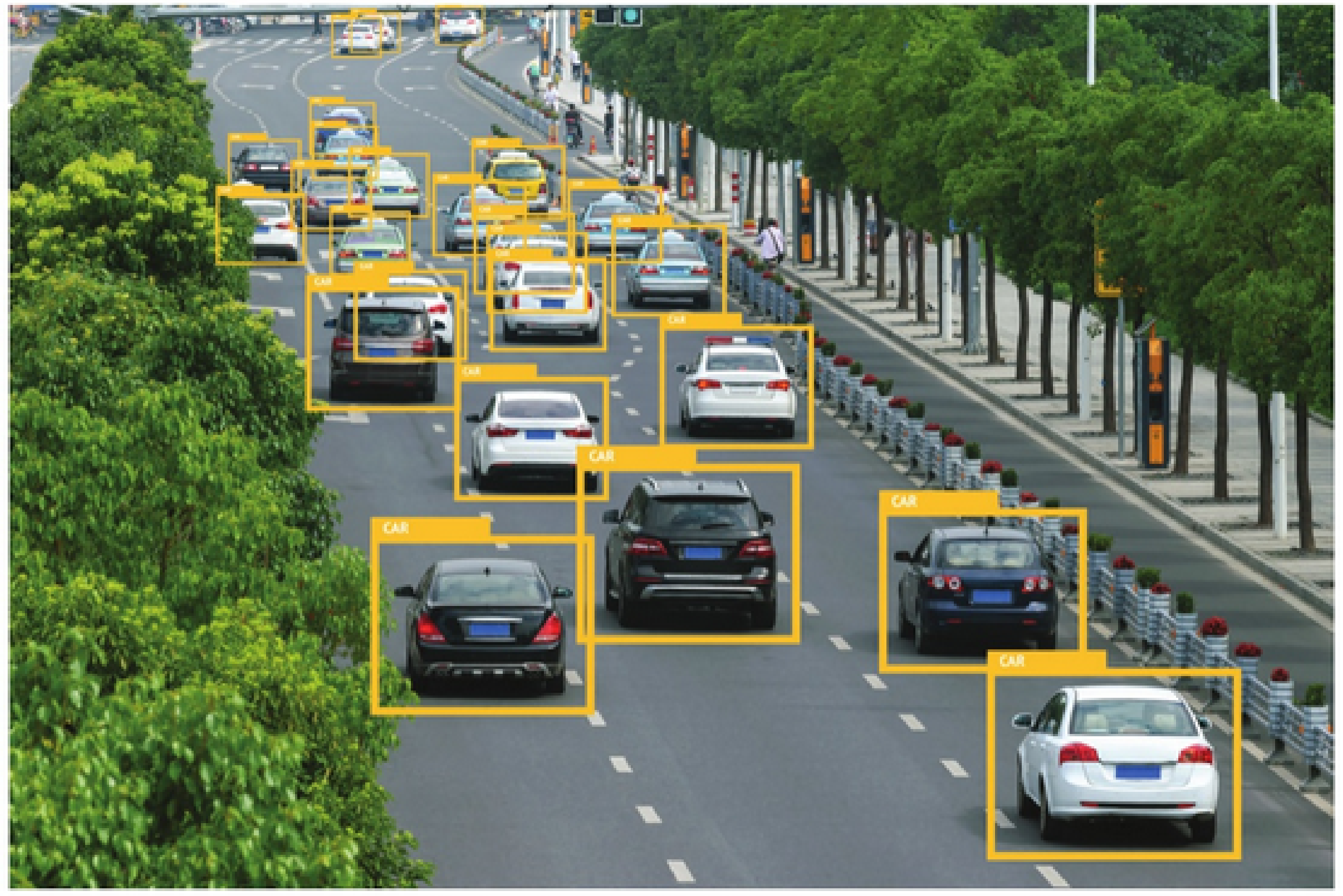
In the email example you studied in the Getting Started activity, the date and time of the email is an example of data that is structured. This is specific data that has a format. It could easily be used to search for the email if it was stored in a database.

Another common use of structured data is in **machine learning**. Machine learning relies on data being structured so that it can easily be searched and manipulated, improving the learning of the artificial intelligence (AI) that uses it.

For example, a plant identification app could use machine learning. Lots of structured data about different plants could be stored by the app, such as:

- plant colour
- plant height
- plant type.

This is data that is specific and can have a set format. The app can use this structured data and learn more about plants from it.



▶ Structured data about cars could be stored to allow machine learning to learn and identify different cars

UNSTRUCTURED DATA

Unstructured data is data that does not have a set structure or format. It could contain a mixture of lots of different data types. Unstructured data is often **qualitative** rather than **quantitative**.

In the email example you studied in the Getting Started activity, the message content of the email is an example of data that is unstructured. If this data was stored in a database, it would be difficult to use it to search for something specific.

Another example of unstructured data are the social media posts that you may regularly create. They will probably have no set structure and will just contain information for your friends about the thing that you are doing that day.

KEY TERMS

qualitative data: data that cannot be easily measured, such as written descriptions

quantitative data: data that can be measured, such as a set of numeric values

unstructured data: data that has no specific format

Often, when companies want marketing information about you, they have to decide whether they want to ask for this in a structured or unstructured format. The benefit of asking for data in a structured format is that it is easy for them to analyse it and use the results to inform their marketing decisions. The benefit of asking for data in an unstructured format is that they give the customer the freedom to provide whatever information they want to as their response. However, this can then be very difficult to analyse, as it has no set format.

For example, a mobile phone company wants to do some customer research about what a customer's favourite feature is of the new phone that they have released. They could ask customers to give this as structured data or unstructured data.

As structured data they could ask:

What is your favourite feature about your new phone? Select one option:

☐ Screen size
☐ Battery life
☐ Screen resolution
☐ Processor speed
☐ Camera quality

As unstructured data they could ask:

Describe your favourite feature of your new phone.

SKILLS
INTERPRETATION

ACTIVITY 1

For the phone features listed above:

1. Why is the first example an example of structured data?
2. Why is the second example an example of unstructured data?
3. What would be the benefit to the phone company of using the structured example?
4. What would be the drawback to the phone company of using the structured example?
5. What would be the benefit to the phone company of using the unstructured example?
6. What would be the drawback to the phone company of using the unstructured example?

SKILLS
ANALYSIS, CRITICAL THINKING

ACTIVITY 2

Copy this table.

Features of structured data	Features of unstructured data

Write at least three features for each type of data. You may want to use the internet to do further research.

KEY TERM

data extraction: the process of collecting data from various sources and processing it to retrieve meaningful information

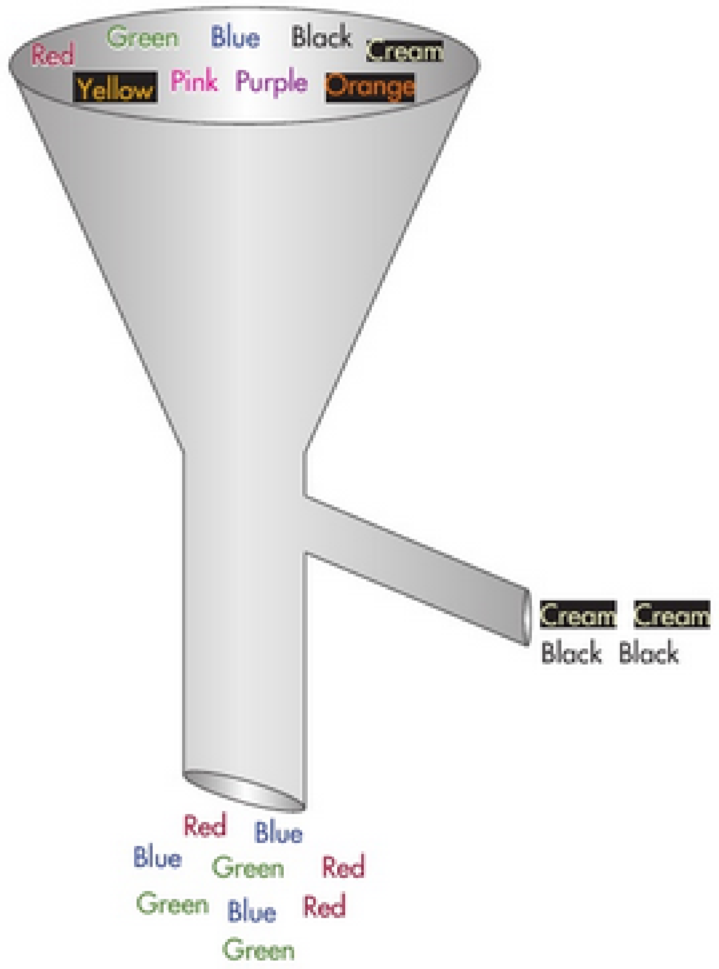
DATA EXTRACTION AND ITS BENEFITS

Data extraction is the process of taking the most beneficial data from a large set of data and processing it to create meaningful information. This information can then be used by an organisation to improve their daily actions and procedures, to create more sales or better services.

Companies often carry out a process called data mining. This will result in a lot of data being collected for the organisation. They will need to process this data to get some meaningful information from it that they can use.

For example, a clothing company wants to release a new sweatshirt. They want to release it in five different colours. They know they want to have black as one colour and cream as another, but they do not know what three other colours would sell the best. They think it might be blue, green and red, but are not sure.

The company could mine data to find out the favourite colour of every customer that has filled in a customer survey for a clothing company. This would probably result in millions of responses about people's favourite colour. They could then process this data by removing the black and cream responses, then request the next three colours with the highest number of responses. This would allow them to see if blue, green and red would be the correct choices, or if there are any other colours that may be more popular.



▶ Figure 5.1 An illustration of data mining to find out the best five colours to use for a new sweatshirt

The information that is revealed as a result of data extraction is very valuable to an organisation. It can help inform lots of decisions such as:

- what their next product should be
- how to improve the sales of a current product
- what investments to make
- what services they could improve
- what the next technological advances could be
- how they are performing against their competitors.

This information could then be used to make important decisions about things such as marketing campaigns, new product development and where to base a new organisation.

KEY TERMS

field: an individual piece of data about an item that is stored in the database; it is a column in the database

record: a collection of all the fields about one item in the database; it is a row in the database

table: a collection of data that is related and structured in rows and columns

STRUCTURED DATA IN A DATABASE

One of the main ways that data is structured is in a database. A simple database normally has one **table**. A table in a database is a collection of data that is related. It is normally structured in columns and rows. A column in a database is called a **field**. Each field is an individual piece of data about an item that is stored in the database. A row in a database is called a **record**. A record is a collection of all the fields about one item in the database. Table 5.1 shows an example of a field and a record in a table.

TABLE 5.1: AN EXAMPLE OF A FIELD AND A RECORD

Song title	Artist	Genre
Happy in love	T. Singer	Pop
Dance all day	The Dances	Dance
Rock and rock	The 2 Rs	Rock
Let's pop!	T. Singer	Pop

← Example of a record

Example of a field

SKILLS
INTERPRETATION

ACTIVITY 3

How many fields are in the database in Table 5.1?

How many records are in the database in Table 5.1?

KEY TERMS

composite key: two fields in a table that are combined to create a unique identifier for the table

foreign key: a primary key that appears in another table in the database to create a link between the two tables

primary key: a field in a database that is used as a unique identifier for a record; this is because the data stored in the field is unique for each record

relational database: a database that has multiple tables that are linked together using certain fields as keys

A more complex database will have multiple tables, and this type of database is called a **relational database**.

Types of keys in a database

The tables in a relational database are linked together. This is done through the use of keys. There are three different types of keys that can be used to do this:

- **primary key**
- **foreign key**
- **composite key**.

Every database should have a primary key. This is a field in the database where every piece of data that is stored in that field for each record is unique. This makes it a unique identifier for each record in the database. The database in Table 5.1 did not have a primary key. This is because it is possible for each field to have repeating data. Two songs could have the same name. The same artist is listed twice. Many songs could have the same genre. A primary key can be added to this table to make sure that each record has a unique identifier. Table 5.2 shows the database with a Song ID field added that can be used as the primary key.

TABLE 5.2: DATABASE WITH A PRIMARY KEY (THE SONG ID) ADDED

Song ID	Song title	Artist	Genre
001	Happy in love	T. Singer	Pop
002	Dance all day	The Dances	Dance
003	Rock and rock	The 2 Rs	Rock
004	Let's pop!	T. Singer	Pop

Each piece of data in the Song ID field that has been added to the database is unique. Multiple records cannot have the same song ID. This means that it is a unique identifier for each record. The benefits of having a primary key in a database are that:

- it helps prevent duplicate records being created in a database
- it helps with updating or deleting a specific record
- it helps set up relationship links between different tables in a database.

It is the last benefit that is used when creating a relational database. One method of linking two tables together in a relational database is by taking the primary key in one table in the database and adding it to another table in the database. When a primary key in one table is used in another table in the database it is called a foreign key. For example, another table is added to the database that contains the contact details for each artist. This table is given a primary key. The primary key from this table can then be used as a foreign key in the Songs table. Tables 5.3 and 5.4 show how a primary key is used as a foreign key to link two tables in a relational database.

TABLES 5.3 AND 5.4: PRIMARY AND FOREIGN KEYS

Artist ID	Artist	Contact number
ART01	T. Singer	0110 1111111
ART02	The Dances	0220 2222222
ART03	The 2 Rs	0330 3333333

Primary key → Foreign key

Song ID	Song title	Artist ID	Genre
001	Happy in love	ART01	Pop
002	Dance all day	ART02	Dance
003	Rock and rock	ART03	Rock
004	Let's pop!	ART01	Pop

By using the primary key from the Artists table as a foreign key in the Songs table, a link is created between the two tables. This link means that all the data stored about the artist in the Artists table doesn't need to appear in the Songs table. Linking tables in this way is called creating a relationship between the tables. You will learn more about different types of relationships between tables later in this topic.

The third type of key that can be created is a composite key. A composite key is another way of creating a unique identifier for each record in a table. You previously added the Song ID field to the Songs table to create a primary key. You can also create a unique identifier by combining two or more fields to create a composite key. It is very unlikely that an artist will release a song with the same title twice. This means that the song title field and the artist field could be combined to create a composite key. Table 5.5 shows an example of a composite key.

TABLE 5.5: AN EXAMPLE OF A COMPOSITE KEY

Song title	Artist	Genre
Happy in love	T. Singer	Pop
Dance all day	The Dances	Dance
Rock and rock	The 2 Rs	Rock
Let's pop!	T. Singer	Pop

← Composite key →

SKILLS

INTERPRETATION, CREATIVITY

ACTIVITY 4

Create an infographic that explains the three different types of key that can be created in a relational database.

Share your infographic with a partner. Ask them to give you feedback about two things that they understand well from your infographic. Ask them to give you feedback about one thing they think could be explained more clearly in your infographic.

KEY TERMS

entity: another name for a single table in a database

one-to-one relationship: a link where one record in a table is related to only one other record in another table

relationships: a link that is created between two tables in a database

CREATING RELATIONSHIPS BETWEEN ENTITIES

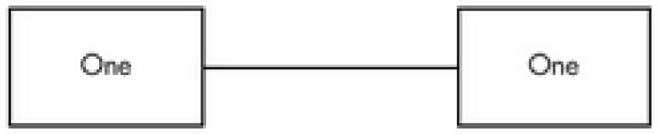
Another name for a table in a database is an **entity**. There are three different types of **relationships** that can be built between entities in a database:

- one-to-one
- one-to-many
- many-to-many.

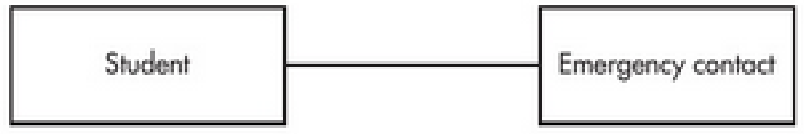
One-to-one relationship

A **one-to-one relationship** is a link where one record in an entity is related to just one record in another entity. For example:

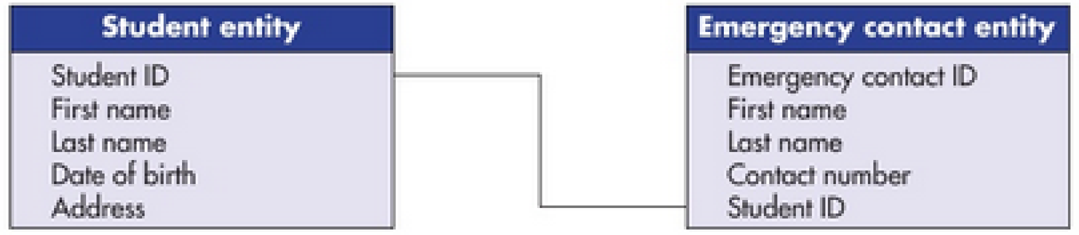
A school database may store the details of students in one table and the details of their emergency contact in another table. The record for one student in the student entity will only be linked to one record in the emergency contacts entity. This is because one student only has one emergency contact. A one-to-one relationship can be shown as a diagram:



In the example it would be:



A one-to-one relationship is normally created by linking a primary key and a foreign key. In this example, the student has a student ID that is the primary key in the student table. This is then added to the emergency contact table as a foreign key to link the two tables.



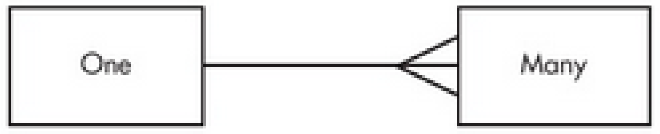
KEY TERM

one-to-many relationship: a link where one record in a table is linked to many other records in another table

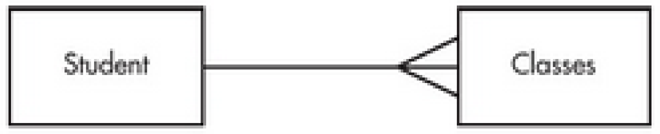
One-to-many relationship

A **one-to-many relationship** is a link where one record in an entity is related to multiple records in another entity. For example:

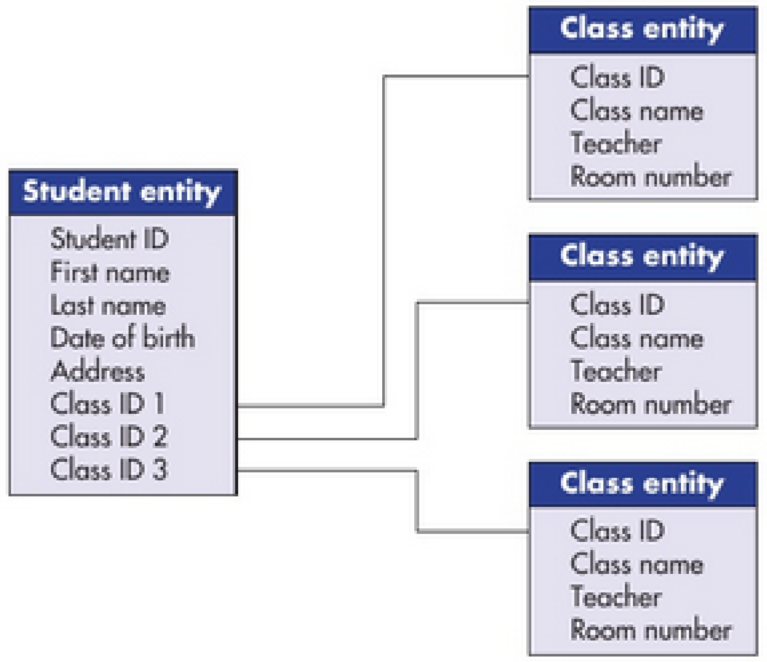
The school database may store the details of students in one table and the details of classes the student can take in another table. The record for one student will be linked to multiple records in the classes entity. This is because one student can take multiple classes. A one-to-many relationship can be shown as:



In the example it would be:



A one-to-many relationship is also created by linking the primary key in one table to a foreign key in another table. If a student takes three different classes, the student record is linked to three different records in the class entity.


Student entity
Student ID
First name
Last name
Date of birth
Address
Class ID 1
Class ID 2
Class ID 3
Class entity
Class ID
Class name
Teacher
Room number
Class entity
Class ID
Class name
Teacher
Room number
Class entity
Class ID
Class name
Teacher
Room number

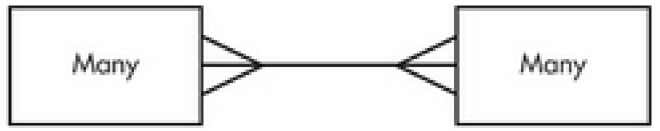
KEY TERM

many-to-many relationship: a link where many records in one table can be linked to many records in another table; this relationship normally has a middle table that creates two one-to-many relationships between the tables instead

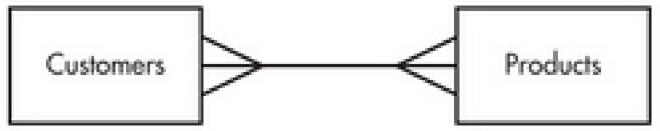
Many-to-many relationship

A **many-to-many relationship** is a link where multiple records in one entity are related to multiple records in another entity. For example:

A company stores details about its customers and its products. It stores customers' details in one table and product details in another table. Many customers can buy many products. Multiple records in the customer entity can be linked to multiple records in the products entity. A many-to-many relationship can be shown as:

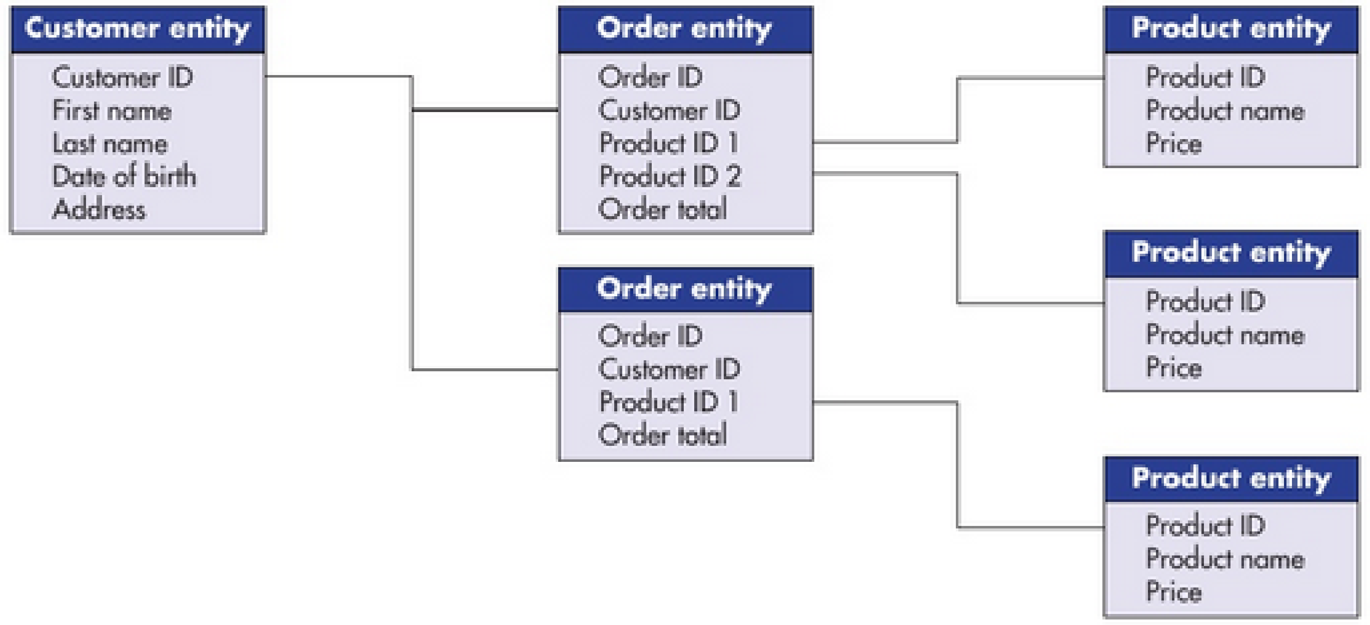


In the example it would be:


Customers
Products

In a relational database, you would not create a many-to-many relationship. This is because it would not make for good record keeping. It would be very difficult to keep track of which customers had ordered which products, as multiple customers can be linked to multiple products.

To try and resolve this issue, a many-to-many relationship is normally broken up into two one-to-many relationships. This is normally done by using a third table called a joining entity. In this example a joining entity could be an order table. One customer can place many orders and one order can contain many products. This would create a joining entity between the two other entities that can also have a unique ID as a primary key. The primary keys from the other two entities can then be used as foreign keys in the joining entity.


Customer entity
Customer ID
First name
Last name
Date of birth
Address
Order entity
Order ID
Customer ID
Product ID 1
Product ID 2
Order total
Order entity
Order ID
Customer ID
Product ID 1
Order total
Product entity
Product ID
Product name
Price
Product entity
Product ID
Product name
Price
Product entity
Product ID
Product name
Price

SKILLS
INTERPRETATION, CREATIVITY

ACTIVITY 5

Draw an entity relationship diagram to represent the relationship between the table containing the artists' information and the table containing the song information shown in Tables 5.3 and 5.4.

Another table is added to the database that stores information about venues where the artists play. Several artists can play at a venue and an artist can play at several venues. Draw an entity relationship diagram to represent the relationships for the new table.

KEY TERMS

query: a search that is created to find specific data in a database

structured query language (SQL): the programming language that is used to create a query

STRUCTURED QUERY LANGUAGE (SQL)

One of the main reasons to store data in a database is so that you can search through it at a later date to find the data that you need. To search for particular data in a database you need to create a **query** for the database. This is like asking the database a question, such as 'How many students take ICT as a class?'

To create a query for a database you use a language called **structured query language (SQL)**. This has lots of different commands that can be used to create different kinds of searches.

To learn the different SQL commands, you will start by using the simple song database you used in Table 5.2. This table is called the Songs table.

Song ID	Song title	Artist	Genre
001	Happy in love	T. Singer	Pop
002	Dance all day	The Dances	Dance
003	Rock and rock	The 2 Rs	Rock
004	Let's pop!	T. Singer	Pop

SELECTING DATA FROM A TABLE

The first commands you need for retrieving data are the SELECT and FROM commands. These commands are used to select a field to be output from a table. The syntax of the query is:

```
SELECT field FROM table;
```

For the Songs table this would be:

```
SELECT Song title FROM Songs;
```

The output of this query would be:

```
Happy in love
Dance all day
Rock and rock
Let’s pop!
```

These commands can also be used to select multiple fields to be output from a table. For example:

```
SELECT Song title, Genre FROM Songs;
```

The output of this query would be:

```
Happy in love    Pop
Dance all day    Dance
Rock and rock    Rock
Let’s pop!       Pop
```

You can use these commands to output all the fields from a table. You could do this by referring to each field name, but there is a quicker way to do this by using the * symbol. The * symbol means select all fields. For example:

```
SELECT * FROM Songs;
```

The output of this query would be:

```
001    Happy in love    T. Singer     Pop
002    Dance all day    The Dances    Dance
003    Rock and rock    The 2 Rs      Rock
004    Let’s pop!       T. Singer     Pop
```

SKILLS
INTERPRETATION, PROBLEM SOLVING

ACTIVITY 6

Write an SQL query that would output the artist and the genre for all records from the Songs table.

Share your query with a partner. Are they identical? Would they both work?

You may want to be more specific about the data that you are searching for. You may not want all the records for a field to be displayed; you may want only the records that contain specific data. To create a query like this you will need to add the WHERE command to the query. For example:

```
SELECT Song title, Artist FROM Songs WHERE Artist = ‘T. Singer’;
```

The output of this query would be:

```
Happy in love    T. Singer
Let’s pop!       T. Singer
```

This query is used to show us all the songs in the database that have the artist T. Singer.

SKILLS
INTERPRETATION, PROBLEM SOLVING

ACTIVITY 7

Write an SQL query that would output the song title and artist for any song that has the genre of pop.

Song ID	Song title	Artist	Genre
001	Happy in love	T. Singer	Pop
002	Dance all day	The Dances	Dance
003	Rock and rock	The 2 Rs	Rock
004	Let’s pop!	T. Singer	Pop

Write down the expected output of your query.

You may need to find data in a database that is above or below a value rather than equal to a value. We will add another field showing the number of sales for each song to the database to learn this example.

Song ID	Song title	Artist	Genre	Number of sales
001	Happy in love	T. Singer	Pop	10 000
002	Dance all day	The Dances	Dance	15 000
003	Rock and rock	The 2 Rs	Rock	8 000
004	Let’s pop!	T. Singer	Pop	12 000

You can now create a query to output the song title, artist and number of sales for any song that has more than 10 000 sales. For example:

```
SELECT Song title, Artist FROM Songs WHERE Number of sales > 10000;
```

The output of this query would be:

```
Dance all day    The Dances
Let’s pop!       T. Singer
```

This type of query is not restricted to numerical fields – you can also use it with text fields. For example, you may want to find all the songs in the database that begin with a letter between A to M inclusive in the alphabet. The query for this would be:

```
SELECT Song title FROM Songs WHERE Song title <= ‘M’;
```

The output of this query would be:

```
Happy in love
Dance all day
Let’s pop!
```

There may be cases where you want to search for more than one condition in the data. To add conditions to searches you can use the commands AND, OR or NOT.

The AND command means that the query will output any record that meets both conditions.

The OR command means that the query will output any record that meets either of the conditions.

The NOT command means that the query will output any record that does not meet the stated condition.

For example:

```
SELECT Song title FROM Songs WHERE Artist = ‘T. Singer’ AND Genre = ‘Rock’;
```

This would not output any records, as there are no records that have the artist as T. Singer and the genre as Rock.

```
SELECT Song title FROM Songs WHERE Artist = ‘T. Singer’ OR Genre = ‘Rock’;
```

The output for this would be:

```
Happy in love
Rock and rock
Let’s pop!
```

These are the song titles that either have the artist T. Singer or have the genre of Rock.

```
SELECT Song title FROM Songs WHERE NOT Genre = ‘Rock’;
```

The output for this would be:

```
Happy in love
Dance all day
Let’s pop!
```

These are the song titles that have any genre apart from Rock.

SKILLS

INTERPRETATION, PROBLEM SOLVING

ACTIVITY 8

Write an SQL query that outputs the song ID and number of sales for any song that has 10 000 or fewer sales or has the genre of dance.

Write down the expected output of your query.

INSERTING, AMENDING AND DELETING RECORDS

SQL can also be used to edit a database by inserting, amending and deleting records in the database.

To insert a record into a table you would use the INSERT INTO command. To insert a record, the SQL query needs to refer to the table name and give the value to insert for each field in that table. For example:

```
INSERT INTO table VALUES (value1, value2, value3, value4, value5);
```

To insert the new song ‘Forever’ by The 2 Rs into the table, you would use the command:

```
INSERT INTO Songs VALUES (005, ‘Forever’, ‘The 2 Rs’, ‘Rock’, 5000);
```

The table will now be:

Song ID	Song title	Artist	Genre	Number of sales
001	Happy in love	T. Singer	Pop	10 000
002	Dance all day	The Dances	Dance	15 000
003	Rock and rock	The 2 Rs	Rock	8 000
004	Let’s pop!	T. Singer	Pop	12 000
005	Forever	The 2 Rs	Rock	5 000

EXAM TIP

When using text values in a condition, for example a word, the text needs to be in single quotation marks for SQL to recognise it as text. When using numeric values in a condition, the number should ***not*** be in quotes as you do not want SQL to treat it as text. So in the example:

```
INSERT INTO Songs VALUES (005, ‘Forever’, ‘The 2 Rs’, ‘Rock’, 5000);
```

‘Forever’, ‘The 2 Rs’ and ‘Rock’ need to be in quotes, but 005 and 5000 do not.

To amend a record in a table, you use the UPDATE, SET and WHERE commands. You need to refer to the table, select the field that you want to update and select the data you want to update in that field. For example:

```
UPDATE table SET field and condition WHERE the data that should be changed
```

In the Songs database, the artist The 2 Rs recruit another member to their band. They are now called The 3 Rs. Their name needs to be updated in the database.

The command to do this is:

```
UPDATE Songs SET Artist = ‘The 3 Rs’ WHERE Artist = ‘The 2 Rs’;
```

This would update all the entries of The 2 Rs in the artist field to The 3 Rs. The table will now be:

Song ID	Song title	Artist	Genre	Number of sales
001	Happy in love	T. Singer	Pop	10 000
002	Dance all day	The Dances	Dance	15 000
003	Rock and rock	The 3 Rs	Rock	8 000
004	Let’s pop!	T. Singer	Pop	12 000
005	Forever	The 3 Rs	Rock	5 000

To delete a record from the table, you use the DELETE FROM and WHERE commands. You need to refer to the table and the field and condition for the record that you want to delete. For example:

```
DELETE FROM table WHERE field and condition
```

The song ‘Forever’ is no longer available for sale. The record for this song needs to be removed from the database. It is possible that another song in the database is also called Forever, so the best file to use to delete this record is the unique identifier for the record, which is the song ID.

The command to do this is:

```
DELETE FROM Songs WHERE Song ID = 005;
```

This would remove all the data for record 005 from the table.

SKILLS

INTERPRETATION

ACTIVITY 9

Write an SQL query to insert a new record into the Songs table.

Write an SQL query to amend the sales for the song ‘Let’s pop!’ to 12 500.

Write an SQL query to remove the song ‘Rock and rock’ from the table.

Exchange your queries with a partner. Use your partner’s queries to amend your table. Do your tables look the same after the queries have been run?

CREATING SEARCHES FOR DATA IN MULTIPLE TABLES

```
SELECT
    OrderH.invoiceNo, OrderH.invoiceDate,
    OrderD.itemCode, I.itemName,
FROM
    OrderHeader AS OrderH
    INNER JOIN Customer AS Cust ON
    INNER JOIN OrderDetail AS OrderD ON
    INNER JOIN Item AS I ON OrderD.itemCode
WHERE
    OrderD.netPrice > 1000
ORDER BY
    OrderH.customerCode, OrderD.netPrice
```

▶ Structured query language can be used to search in multiple tables

The example data in this section are the two tables that you used earlier in this topic for songs and artists (see Tables 5.3 and 5.4).

Songs table

Song ID	Song title	Artist ID	Genre
001	Happy in love	ART01	Pop
002	Dance all day	ART02	Dance
003	Rock and rock	ART03	Rock
004	Let's pop!	ART01	Pop

Artists table

Artist ID	Artist	Contact number
ART01	T. Singer	0110 1111111
ART02	The Dances	0220 2222222
ART03	The 2 Rs	0330 3333333

You might want to create a search to display data from more than one table. For example, you might want to find the name of the artist for a particular song ID. Those two pieces of data are stored in different tables. For this search you would need to use the SELECT, FROM and INNER JOIN ON commands. For example:

```
SELECT table.field FROM first table INNER JOIN second table ON fields that have
matching data in both tables
```

To find the name of the artist for a particular song ID, this would be:

```
SELECT Songs.SongID, Artists.Artist FROM Songs INNER JOIN Artists ON Songs.
ArtistID=Artists.ArtistID;
```

The output for this would be:

```
001   T. Singer
002   The Dances
003   The 2 Rs
004   T. Singer
```

You might need to get the data from one column in one table and join it to the data from another table to create a single output for that data, for example to display a single list of all the song IDs and artist IDs that are stored in the two tables. To do this you would use the SELECT, FROM and UNION commands. For example:

```
SELECT field FROM table UNION SELECT field FROM table
```

Using the example this would be:

```
SELECT Song ID FROM Songs UNION SELECT Artist ID FROM Artists;
```

The output for this would be:

```
001
002
003
004
ART01
ART02
ART03
```

COUNTING, GROUPING AND ORDERING RECORDS

To count the number of records in a database, or to count the number of records that meet a certain condition, you use the SELECT, COUNT, FROM and WHERE commands. For example:

```
SELECT COUNT (*) FROM Songs
```

This would output the value 4, as there are four records in the Songs table.

```
SELECT COUNT (*) FROM Songs WHERE Genre = 'Pop'
```

This would output the value 2, as there are two records in the Songs table where the genre is Pop.

There may be times when you need to group the data in a field and find out how many times that data is repeated in a field. For example, if you wanted to group the songs by genre and count the number of songs in each genre, you would use the SELECT, COUNT, FROM and GROUP BY commands. For example:

```
SELECT COUNT (field to count), field to search FROM table GROUP BY field
```

For the grouping by genre example, this would be:

```
SELECT COUNT (Song ID), Genre FROM Songs GROUP BY Genre
```

The output for this would be:

```
2    Pop
1    Dance
1    Rock
```

You can also order the output. The current order is by count from highest to lowest, but this is coincidental and hasn't been done purposely. If you wanted to purposely order from highest to lowest by the count, you would add the ORDER BY and the DESC commands. ORDER BY puts the data into ascending order and DESC puts it into descending order. For example:

```
SELECT COUNT (Song ID), Genre FROM Songs GROUP BY Genre ORDER BY
COUNT (Song ID) DESC;
```

To sort the output in lowest to highest order, you would use ASC for ascending rather than DESC.

You could also sort the data in the genre field into alphabetical order. To do this, you order it by the field name. For example:

```
SELECT COUNT (Song ID), Genre FROM Songs GROUP BY Genre ORDER BY Genre;
```

The output for this would be:

```
1    Dance
2    Pop
1    Rock
```

SKILLS

INTERPRETATION, PROBLEM SOLVING

ACTIVITY 10

Write an SQL query that outputs the name of an artist and the number of songs they have. Order the output by the count from highest to lowest.

WILDCARD SEARCHES

There may be times when you do not want to search for a single specific piece of data – your search criteria may be more vague. To create this kind of search you can use wildcards in your searches. There are two wildcard symbols: % and _. The % wildcard symbol represents any number of characters in the data and the _ symbol represents one character in the data.

If you wanted to find the records in the artist table for any artist that begins with the letter T, the query would be:

```
SELECT * FROM Artists WHERE Artist LIKE T%
```

The output for this would be:

```
ART01        T. Singer       0110 1111111
ART02        The Dances      0220 2222222
ART03        The 2 Rs        0330 3333333
```

This is because all three of the artists have names that start with the letter T.

To find display artists that have a name ending in the letters 'es', the query would be:

```
SELECT * FROM Artists WHERE Artist LIKE %es
```

The output for this would be:

```
ART02                  The Dances     0220 2222222
```

You can use the % wildcard to replace any number of letters in the data stored in the field. If you want to use a wildcard to replace a specific number of letters for the data in the field, you would use the _ wildcard.

For example, to find out which records have a genre that is three letters long and begins with a P, the query would be:

```
SELECT * FROM Songs WHERE Genre LIKE P__;
```

This would find any data in the genre field that starts with P and has any two characters after it. This is because there are two _ symbols after the P.

The output for this would be:

```
001    Happy in love    ART01    Pop
004    Let's pop!       ART01    Pop
```

EXAM TIP

When you are using any field names, table names and data stored in fields in an SQL query, you must make sure that the spelling is exact and the case is exact. If the field name is StudentID but you use studentID, it will not work, as the field name is not exact.

Table 5.6 contains a summary of the commands you have learnt in this topic.

TABLE 5.6: A LIST OF SQL COMMANDS

SQL command	Description
SELECT	Used to select records from one or more tables in the database
FROM	Specifies which table you want to select or delete data from
WHERE	Used to filter records for certain criteria
AND	Allows multiple criteria to be used in a search and will find all records that meet all criteria
OR	Allows multiple criteria to be used in a search and will find all records that meet at least one of the criteria
NOT	Allows records that contain certain criteria to be discarded from the search results
INSERT INTO	Used to insert a new record into a table
VALUES	Allows a set of values to be specified with other commands such as the values to insert into a table
UPDATE	Used to change an existing record in a database
SET	Specifies which data should be updated in the database
DELETE	Used to delete a record from a database
INNER JOIN	Selects records from multiple tables in a database that have matching values in the tables
ON	Used to define the join condition for the specific data where the tables should be joined
UNION	Allows two or more sets of results from SELECT searches to be combined
COUNT	Counts the number of records that match a certain criteria
GROUP BY	Used to group records in a database by one or more fields
ORDER BY	Used to order a set of results from a search
DESC	Orders results in descending order
ASC	Orders results in ascending order
LIKE	Searches for a specific pattern in the data and is used with wildcard searches

THINKING LIKE AN IT SPECIALIST

Many companies use huge databases as part of their daily operations. The data stored in these databases is very valuable to the company. They also have a duty to keep it safe and up to date. You learnt about how to use SQL to update the data in a database. Why do you think it is important for a company to keep their data up to date?

CHECKPOINT

1. Explain the difference between structured and unstructured data.
2. What are the benefits of data extraction?
3. What is the difference between a field and a record?
4. Describe the purpose of a primary key.
5. What is a composite key?
6. Explain how one-to-many relationships work.
7. What is SQL?
8. What is a query?
9. What is the purpose of the SELECT command?
10. Describe the function of the UPDATE command.
11. Explain how a wildcard is used.

SKILLS

INTERPRETATION, PROBLEM SOLVING, CRITICAL THINKING

EXAM PRACTICE

1. An office supplies company stores information about its customers and products.

(a) The data stored by the office supplies company is structured data.

(i) Describe what is meant by structured data. [2 marks]

(ii) Give one reason why the office supplies company uses structured data instead of unstructured data. [1 mark]

(b) The office supplies company database includes the tables tbl_Customers and tbl_Orders. Sample data is shown in these tables.

tbl_Customers

CustomerID	FirstName	LastName	Address	ContactNumber	Email
CUST001	Ona	Li	10 1st Street	0110 1111111	onali1@email.com
CUST002	Pradeep	Patel	12 2nd Street	0220 1212121	ppatel@email.com
CUST003	Yuri	Akoye	14 3rd Street	0330 3333333	ya99@email.com
CUST004	Emily	Jones	16 4th Street	0440 1231234	emj10@email.com

tbl_Orders

OrderID	ProductID	Quantity	OrderTotal	CustomerID
ORD1001	PROD201	1	15.95	CUST001
ORD1002	PROD150	4	34.80	CUST002
ORD1003	PROD150	2	17.40	CUST001
ORD1004	PROD919	4	67.20	CUST003

(i) Give the name of a primary key in the database. [1 mark]

(ii) Give the name of a foreign key in the database. [1 mark]

(iii) Write an SQL query that will display the customer's first name, last name and email address. [2 marks]

(iv) Write an SQL query that will display all the fields for any orders that have four or more products ordered and have a total of more than $50. [4 marks]

(v) Write an SQL query that:

- displays the customer ID, customer first name, order ID and order total
- orders the output from highest to lowest for the order total. [5 marks]

TOPIC 6 WIDER ISSUES

The IT systems that we use are having a huge and lasting impact on our environment. It is vital that we know what they affect and how we can make a difference to this. We have created smart homes and smart cities using technology. In this topic, you will learn about the environmental impact of creating technological equipment, using it and disposing of it. You will learn how technology can be used to monitor our environment and how we can use this as an advantage, especially when trying to use precious resources more efficiently. You will also learn about some legal, moral and ethical issues surrounding the use of technology.

TOPIC 6 WIDER ISSUES

LEARNING OBJECTIVES

By the end of this topic you should be able to understand:

- the environmental impact of creating, using and disposing of technological equipment
- the positive impact of using technology on the environment when used for monitoring and using resources efficiently
- the legal, moral and ethical issues connected to the use of IT systems
- the impact of wireless access being available everywhere.

GETTING STARTED

You have probably used several different technology devices in your lifetime. Have you ever thought about the impact that your use of technology has on the environment? If you knew that it had negative impacts on the environment, would this make you stop using technology?

▶ The creation, use and disposal of technology has an environmental impact

ENVIRONMENTAL IMPACT OF CREATING TECHNOLOGICAL EQUIPMENT

New and existing technologies are created every day all over the world. The creation of these technologies can have positive and negative effects on the **environment**. We know that the state of our environment is very fragile and, so, it is important to know what these impacts are.

The positive impact of creating technologies comes from the way that we can then use them to monitor lots of processes that affect our environment and make sure that we use **resources** effectively. You will learn more about the positive impact of technology for this purpose later in the topic.

There are several negative impacts connected to the creation of technology. Table 6.1 contains a list of these negative impacts and a description of how they are negatively impacting the environment.

KEY TERMS

environment: the surroundings or conditions in which a person, animal or plant lives or operates

resources: the materials that we have available to us

KEY TERMS

habitat: the natural home of an animal or plant

landfill: a place where non-recyclable waste materials are disposed of and buried in the ground

pollution: the existence of a substance in the environment that has harmful effects on the environment and the things that live in it

TABLE 6.1: NEGATIVE IMPACTS OF CREATING TECHNOLOGY

Impact	Description
Disrupting nature	New factories often need to be built to expand a company's ability to manufacture new and existing technologies. Sometimes the land being cleared to make space for these factories was a natural area where animals were living. This destroys the **habitat** of various living creatures, which can heavily affect the cycle of nature. Fields are cleared of wildflowers that are very important for pollinating insects, such as bees. We need bees for so many of our daily food products and if they were to disappear this would have a huge negative impact on our food system.
Consuming resources	Technologies such as computers, smartphones and tablets contain precious metals such as gold, silver, platinum, palladium and copper. These resources are limited, and they can also leak into water supplies and soil during manufacture and disposal, which causes serious health issues for people and ecosystems living near factories.
Pollution	The manufacturing process for technologies produces air, water and noise **pollution**. Carbon emissions are released into the air during manufacturing. These are greenhouse gases that contribute towards global warming. The gases get released from fossil fuels, such as coal or oil, that are burnt to provide energy in the manufacturing process. You've already learnt how precious metals can leak into water supplies during the manufacturing process. Factories are also often very noisy, and this can cause serious stress and disruption for the people and animals nearby.
Waste	Waste is created during the manufacture of technologies, for example plastic waste from cutting out the plastic cases that are often used for devices. This type of waste is sometimes recycled, but it can also be sent to **landfill** where it will take years to be completely broken down. It can also leak toxic chemicals into the ground.

ENVIRONMENTAL IMPACT OF USING AND DISPOSING OF TECHNOLOGICAL EQUIPMENT

Millions of households and businesses across the world use technology on a daily basis. It is estimated that over 85 per cent of the world's population uses a smartphone. There are both positive and negative effects on the environment connected to the use of technology across the world. Disposal of technology has negative impacts on the environment. The text below shows several positive and negative effects on the environment of the use and disposal of technology.

Positive effects on the environment of using technology

Technology can be used to monitor dangerous and hazardous environments in a factory or around the world. The technology can also automate safety systems, such as sounding alarms or shutting off valves. This is particularly important in places where it is unsafe for humans, such as a nuclear reactor or an active volcano.

We can use technology to create **smart homes**. The technology can be used to monitor and control our use of resources in our houses. Smart meters monitor how much electricity we are using on a daily basis and how long we have our heating/air conditioning systems active for. This can help to reduce the waste of precious resources that are used to generate electricity.

KEY TERM

smart homes: homes containing devices that can be remotely controlled by another device, such as a smartphone

KEY TERM

smart cities: cities that use technology, AI and machine learning to collect data, and monitor and adapt processes in the cities to make them more efficient

Technology can be used to create **smart cities**. It could automatically monitor several different processes that happen in a city, for example to automatically switch streetlights on and off when needed, reducing electricity needs. Technology in smart cities could create more efficient transport systems, for example to filter public transport through busy areas and away from traffic, or make extra resources available, such as opening extra lanes on roads at busy times when there is a lot of traffic. This would reduce carbon emissions, as people may be more likely to use public transport, meaning fewer cars on the road and less traffic stuck in traffic jams.

Technology can be used to create renewable energies such as solar, wind and hydro. This means fewer harmful effects on the environment as there is less need to generate electricity using fossil fuels such as coal and oil.

It is now possible to send messages, stream music and movies, and have meetings – all through the use of technology. Previously, all of these actions would have used lots of resources. Sending letters would have used lots of paper. Listening to music and watching movies generally needed CDs and DVDs. These are made of plastic, which is harmful to the environment, and they require energy to make and to transport. People would use fuel in their cars to drive to work meetings that can now be held online.

Negative effects on the environment of using technology

Although technology can help save lots of resources, it does require resources to operate. The main resource needed is electricity. If fossil fuels such as coal or oil are used to create the electricity, then this has a negative effect on the environment.

A lot of technologies are built into items that release carbon emissions. They are built into things such as cars, planes and factories. All these things increase carbon emissions and contribute towards global warming.

There is a constant push to make technology better, faster and more efficient at what it does. The development and research that goes into this requires many limited resources, such as precious metals, that are then wasted if products are not effective.

People are persuaded by advertising to want the latest technologies, with many people upgrading their technology every year or two. Companies often produce products with 'built-in obsolescence', and are making it harder for people to fix their own devices, so that a device will need to be replaced frequently. Much of the replaced technology is thrown into landfill, which leaks toxic materials into the ground and nearby water sources. It can also spread into food sources and cause health issues. This kind of technology waste is known as e-waste, and sometimes toxic techno trash.

▶ Lots of older technology ends up in landfill; this can cause toxic materials to leak into the soil and into water supplies

SKILLS

INTERPRETATION

ACTIVITY 1

Create a revision quiz that you can use at a later date to test your understanding of the environmental impact of technology.

LEGAL, MORAL AND ETHICAL ISSUES

LEGAL ISSUES ASSOCIATED WITH USING IT SYSTEMS

The use of IT systems can require a large amount of data to be processed, transmitted and stored. This data can be as valuable as personal data and can be fundamental to a person's identity. The data could also be a person's property, such as a photograph that they have taken. This means that data collection, use and transmission needs to be done lawfully, and the data must be kept as safe as possible. There are four main legal areas to consider for this:

- data protection
- intellectual property
- copyright
- computer misuse.

Data protection

Data protection laws are designed to protect data that is collected, stored and transmitted. Most of the data they are designed to protect is data that is collected and stored by companies. Different countries have their own data protection laws. Some countries have very strict data protection laws while other countries have more relaxed laws that are not as comprehensive.

Your data could be collected in your own country, where there are strict data protection laws that mean your data has to be stored securely and used only for necessary circumstances. However, the company that collects your data could store your data in **cloud technology**. This cloud technology could be located in a different country where the laws for storing your data are not as strict. This could lead to your data being stolen or used for unlawful purposes.

KEY TERM

cloud technology: technology that includes servers that provide data storage that you can access from a remote location using the internet

Data protection laws across the world mainly exist to:

- protect the data of an individual
- protect the rights of an individual
- regulate how data is used
- make sure data is stored securely
- make sure only data that is necessary is collected and stored
- define who can gain access to any data stored
- make sure that data that is collected and stored is up to date.

◀ Data collected in one country may be stored in another via cloud technology and may not be fully protected

SKILLS

CREATIVITY, CRITICAL THINKING

ACTIVITY 2

Work with a partner.

- Find out about the data protection laws in your country.
- A company is planning to implement a new system for customers to buy running shoes online. List the personal data that the company will need to collect from each customer. What are the risks for customers if the company does not protect the data properly?
- Share your findings with another group. Did they think of something you didn't? Is the risk the same for each piece of data, or is some of the data more valuable?

KEY TERMS

copyright: a legal form of protection that can be applied to intellectual property; it gives the owner control over what happens with the property and who can use it

intellectual property: property that is the result of creative thinking and actions

Copyright and intellectual property

Copyright is a law that is designed to protect the creators of materials. These materials are often referred to as **intellectual property**. Intellectual property could be:

- a photograph
- artwork
- a video or a movie
- a poem
- a document
- a website
- a song
- a knitting pattern
- an animation
- a video game
- a prototype of an invention
- a performance or a play.

All these things are created by a person or company. The copyright for the creation gives the owner the rights to their intellectual property. This means that they own the right to copy, distribute and adapt their work. It also means that anyone else who wants to do this has to get permission from the owner of the copyright to do so.

Each country has their own copyright laws that govern the copyright of intellectual property created in that country. If a photographer who lives in Bangladesh gets copyright for their photographs in Bangladesh, this same copyright may not apply if the photographs are used in other countries. However, in 1886, an agreement called the Berne Convention was developed. This is an agreement between many countries around the world that consists of a set of legal principles that protect artistic work. These legal principles state what owners of artistic work can control about how their work is used and who can use it.

As well as copyright, there are other types of legal protection that can be obtained for intellectual property, including the following.

Patent – this is normally obtained for new inventions. It stops other people from making or selling another invention that is the same as yours for a period of time.

Trademark – this is a symbol that is used to identify the owner of the brand of a product. For example, many companies can sell their own version of cola drinks, but only Coca-Cola has the trademark for this as it is the recognised brand owner of it. The purpose of a trademark is to identify the source of the product instead of the product itself.

Trade secret – this is a piece of information that is not widely known; it can be sold or licensed. A company may have a trade secret for how they create a product and can prevent this information from becoming common knowledge.

▲ Any intellectual property that has copyright will normally bear the copyright symbol

Computer misuse

Computer-related crime has risen greatly over the last few years. Governments around the world have to work very hard to keep up with the laws and regulations that govern the use of computers in computer-related crimes. Many countries have computer misuse laws that govern the use of computers for this purpose.

Most computer misuse laws aim to govern:

- unauthorised access to information stored on a computer or server (you may know this as hacking)
- unauthorised access with intent to use the computer or information to commit a crime, for example identity theft or fraud
- unauthorised interception of data when it is transmitted from one device to another
- altering or sabotaging data that is stored on a computer or server.

Some countries have found it challenging to prosecute people using their computer misuse laws. This is because many of these laws require intent to be proven; so, there must be substantial evidence that a person intended to hack into a computer. The person in question may claim that they accidentally came across the information.

SKILLS

CREATIVITY, CRITICAL THINKING

ACTIVITY 3

Find out what the computer misuse laws are in your country and what exactly they govern. Create a poster that could be displayed in your classroom to inform the members of your class about the computer misuse laws in your country.

Show your poster to a friend. Compare the information that you have given on your posters. Have they included anything that you have not? Is the information correct on both your posters?

MORAL AND ETHICAL ISSUES ASSOCIATED WITH USING IT SYSTEMS

The laws that you have just learnt about cover what a person legally can and cannot do with an IT system and any intellectual property created. However, there are also moral and ethical issues that arise from the use of IT systems. Morals and ethics are what we discuss as a society about behaviour that we think is right or wrong. Our own ethics and morals are like a code of conduct for living our lives. Morals and ethics can differ from person to person. What you may think is bad behaviour, your friend may think is all right. However, there are some shared moral and ethical issues that most people think of as problematic. Table 6.2 shows several moral and ethical issues, and describes how they are related to the use of IT systems.

KEY TERM

privacy: having a place where you can go that your actions are not seen and monitored, and therefore cannot be judged by others

TABLE 6.2: MORAL AND ETHICAL ISSUES RELATED TO USING IT SYSTEMS

Moral/ethical issue	Description
Privacy	Privacy is a big moral and ethical issue that lots of people have very different opinions about. Having our privacy means having a place that we can be where others are not able to watch and judge our actions. One of the biggest privacy issues with IT systems is the number of technologies that track a user's location. For example, shopping malls may track shoppers coming into and out of the mall. They may use facial recognition technology to do this. Some people may welcome this technology as it can help reduce crime in the mall. However, others may think it is a big invasion of their privacy and may decide not to use the shopping mall as a result. AI needs to collect lots of data about its surroundings to be able to learn and adapt. This can often mean collecting data about any person that it interacts with. One example of this is targeted advertising. AI could collect data about a user's activity online, or it could collect data about their current location. It can then use this data to send targeted adverts to the user about similar things to those they have looked at online, or about products or services that are in the area where they currently are, for example the restaurants that are close by. Some people may welcome this information as they find it helpful. Other people may see it as another big invasion of their privacy, and will not want their activity to be tracked. It is also possible that the content in the targeted advertising is not age appropriate for the person. This is also seen as an ethical issue. One of the biggest issues with privacy is that a lot of this tracking is done without consent. This means that people are not being asked if they agree to being tracked by IT systems such as closed-circuit television (CCTV). CCTV exists around many major cities and is constantly tracking the whereabouts of people all over the world.

(Continued)

KEY TERMS

civil liberties: a person's right to basic freedoms as long as they respect other people's rights

inclusion: the provision of equal access and opportunities to everyone

TABLE 6.2: MORAL AND ETHICAL ISSUES RELATED TO USING IT SYSTEMS (*Continued*)

Moral/ethical issue	Description
Inclusion and association	Inclusion means making sure everybody is included in the use of IT systems and technologies. Some of the bigger moral and ethical issues of inclusion can occur online. The internet has allowed people to set up lots of online communities. These online communities can have many positive effects on society. However, they can also be used to exclude people. This exclusion might be conscious (for example, content that is deliberately targeted to a particular ethnic group) or unconscious, for example assuming that everyone will be celebrating the same festivals, talk about things specific to one culture, or see the world in the same way as they do. Companies have been accused of using AI that has inclusivity issues. It has been argued that AI can be affected by human bias. This means that the opinions and biases of the programmers and the people training the AI get programmed into the AI. Such biases could, for example, perpetuate issues of sexism, racism and ableism. Many people use IT systems to enable association and interaction with others. This is done regularly through online communities, for example religious communities. Most people who are part of these communities share common beliefs and practices that unite them. They are often positive and beneficial places where people can communicate and interact with each other. However, some people argue that interacting in this way can be affected by issues such as extremism. This is when people who portray themselves as part of an online community use manipulative techniques to try to get others to participate in behaviour that can sometimes be dangerous and illegal. Manipulating people through association in this way is seen as highly unethical and a problematic use of IT systems such as the internet. It can also make people associated with the genuine online community very angry when others see their association with the community as extremist activity.
Civil liberties	Governments may limit what people can see using IT systems (like the internet). They may stop people from seeing certain content, for example if it is violent. They may believe they are protecting citizens. However, others believe this is a violation of their civil liberties and that a government should not be allowed to censor what people within a country can see.

(*Continued*)

KEY TERM

digital divide: the divide between people who have access to and can use technology and the internet, and those that do not

TABLE 6.2: MORAL AND ETHICAL ISSUES RELATED TO USING IT SYSTEMS (*Continued*)

Moral/ethical issue	Description
Access	Although millions of people around the world do have access to technology now, there are still many people who do not. There are also people that do not feel comfortable using it. This problem is called the **digital divide**. Services such as buying tickets for public transport and banking services are steadily moving towards IT systems. People who regularly use IT systems may be comfortable using them for this purpose. However, there may be some people, for example elderly people, who have lived much of their life without the use of technology and do not feel comfortable using it for services such as banking. This may make these people feel excluded and that they do not have access to the services that they need. Many people see this as an ethical issue and believe that these necessary services should be accessible to everyone. Another issue is the level of bandwidth that a person has access to. Some people have access to high-speed broadband and can enjoy streaming content such as high-definition movies. However, others who do not have access to high-speed broadband may struggle to stream this kind of content and miss out on the enjoyment. Most of these people are in remote locations or countries that do not have the infrastructure yet to support high-speed broadband. Many think it is an ethical issue that these people are excluded from services as a result.
Accessibility	The accessibility of IT systems refers to the kind of features that are included to make sure that the IT system can be used by everybody, including those who have disabilities. Accessibility features such as the following can be included. • Text-to-speech software – to allow text to be read out loud for users with sight issues. • Dark mode – allows text to be read more easily for users with sight issues and dyslexia. • Speech recognition software – to allow users with limited mobility to control a system using their voice. • Screen magnifier – enlarges the screen so that users with sight issues can see it more easily. • Making font larger – so that users with sight issues can read text more easily. • Keyboard-only navigation – so that users who find it difficult to use a mouse can navigate a system. • Sticky keys – pressing one key at a time instead of having to hold down multiple keys together, for example control, alt and delete or shift and a letter to get a capital; this helps users with mobility issues who may struggle to hold down multiple keys. Creators of IT systems are encouraged to build accessibility features into their designs. If these features are not built into the design and some users are not able to use the system, this may be seen as an ethical issue of inclusivity.

(*Continued*)

TABLE 6.2: MORAL AND ETHICAL ISSUES RELATED TO USING IT SYSTEMS (*Continued*)

Moral/ethical issue	Description
Expression	We all like to express what we think and how we feel about how we want to live our life. Freedom of speech is very important to some people and they see it as an ethical issue if they are limited in this way. Some people feel that they should be able to use IT systems like the internet freely and to post any content that reflects their beliefs. Others think that content that is harmful and offensive to others is an ethical issue and should not be allowed. They believe it should be limited and removed. Many social media platforms do have regulations about the type of content that can be posted. They also limit the age range of people who are allowed access to an account. These measures are taken by the platforms to try and show they are acting ethically regarding this issue.

SKILLS

CRITICAL THINKING

ACTIVITY 4

Imagine you move to a new city. You don't know much about this new city and want to learn a lot about all the cool places that you can go to. You are excited to learn what the city offers.

You get an advert on your social media about an app. The app looks amazing! It will track your location around the city and learn what you like. It will then tell you about all the great places that are nearby. It will also give you information such as whether services like transport and restaurants near you are currently busy.

You download the app and cannot wait to explore the city more. You then start to think a bit more about how the app can give you all this information.

- What do you think are the benefits of the app?
- What do you think are the drawbacks of the app?
- What ethical and moral issues do you think might be connected to the app?

You speak to your friend about this and then decide that you want to make others aware of your thoughts.

Write an article that could be published in your school magazine or a local technology magazine that discusses your thoughts about the three questions you have just considered.

EXAM TIP

Remember, if you are asked a question that requires you to evaluate issues regarding ethics, then you should always write a conclusion. Your conclusion should aim to summarise what you have written. The question may also require you to give your choice between different methods/ideas or suggest how an improvement could be made. You should look at the content of the question to see the important things that you should include in your conclusion.

KEY TERMS

location awareness: devices that are able to determine their own location

wireless access: access to the internet that does not require any cables

THE IMPACT OF WIRELESS ACCESS ON A GLOBAL SCALE

Wireless access is now available all over the world. Many businesses now have free Wi-Fi access available to their customers, and many cities, airports and shopping malls also provide free Wi-Fi access. This has been beneficial to people as it has increased the inclusivity of who has wireless access and all the services and information that can be accessed as a result. However, it has also created several issues. Most of these issues you have already learnt about in the legal, ethical and moral sections of this topic.

One of the main impacts of global wireless access is the amount of information that individuals and companies can now have access to about a person's location. This is called **location awareness**. Lots of the devices that we carry on a daily basis have location awareness abilities. This means that they provide information to others about the physical location of the device that you are carrying.

Location awareness has been a very positive addition to the devices we carry for those who live in unsafe or natural disaster zones. If a natural disaster occurs, such as an earthquake, people are able to use their device to show others their exact location. This can greatly help rescue teams to locate these people and save them much quicker.

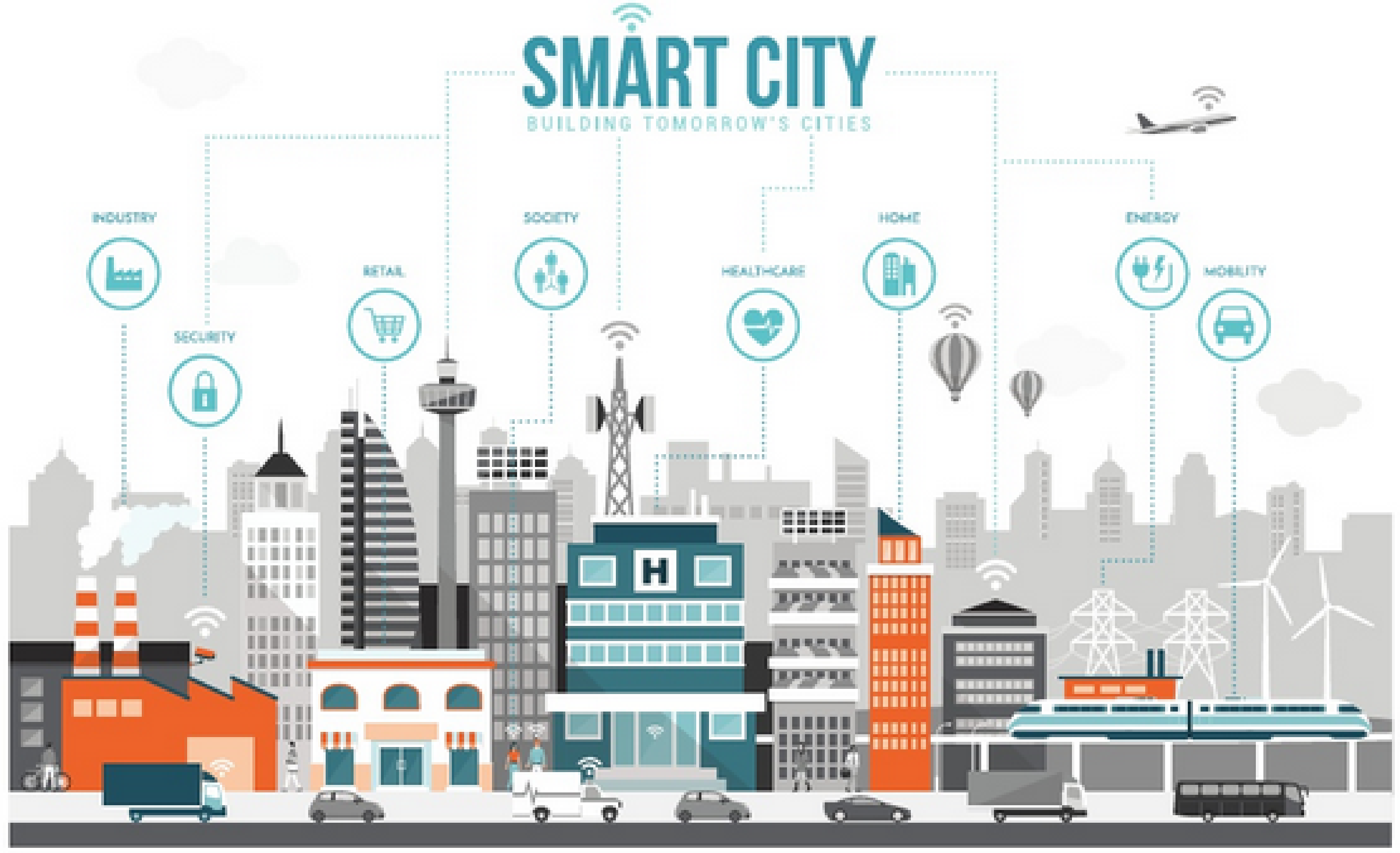
Location awareness has also helped people navigate their way around city and rural areas. GPSs rely on location awareness and allow a user to get directions for where they want to go.

Earlier in the topic you read about privacy issues. One of the main factors in these privacy issues is the use of location awareness. This allows a person to be constantly tracked wherever they go. This may mean they are provided with useful information about their surroundings, but it may also mean that for much of the time their privacy is invaded. It can also mean that their safety could be at risk. If a malicious person wanted to know the whereabouts of someone, they can use this capability to track and find them. As there is wireless access all over the world, this could happen on a global scale.

Location awareness can also be used by companies. Some companies may give devices to their employees that they do not want the employees to be able to use outside of the office. The device can be set to only operate in a certain location, and the device will know from being able to detect its current location whether it is in the office or not. This can prevent employees from removing devices from the office and using them in a way that the company would not want them to be used.

One big development that relies heavily on wireless access is the development of smart cities. Earlier you learnt what a smart city is and some of the ways that IT systems are being used to create smart cities. You also learnt about some of the environmental impacts of this. It is possible to have smart cities through the use of wireless access because it is the most efficient and cost-effective way to connect all the millions of devices in a city to allow it to be a smart city. Smart cities rely on the collection and transmission of huge amounts of data to make the most up-to-date information available about the city. All this information can be transmitted instantly because of wireless access. This means the city can have benefits such as:

- better public transportation
- improved water facilities
- improved heating facilities
- improved lighting facilities
- fewer traffic issues
- safer public spaces
- reduced crime levels.



▶ Figure 6.1 Smart cities can use technology to help provide beneficial services for those who live in and visit the city

SKILLS

CREATIVITY, CRITICAL THINKING

ACTIVITY 5

Create an infographic to show how the use of technology in smart cities is having a positive effect on the environment.

THINKING LIKE AN IT SPECIALIST

AI is increasingly used to monitor and control aspects of our lives. It is used in smart homes and smart cities to help make decisions about how precious resources can be saved and used effectively. How do you feel about AI being used in this way? Do you see more benefits or drawbacks in the use of AI for this purpose?

CHECKPOINT

1. What are the ways in which the environment can be affected by IT systems?
2. How can IT systems cause pollution?
3. How are precious metals linked to IT systems?
4. What is a smart home and how does it make use of IT systems?
5. List the things that data protection laws govern.
6. Explain what is meant by copyright.
7. How is privacy affected by IT systems?
8. What ethical issues can occur with access to IT systems?
9. What is a smart city and how does it make use of location awareness?

SKILLS

ANALYSIS, CRITICAL THINKING, INTERPRETATION

EXAM PRACTICE

1. A company is concerned about the impact the technology it uses is having on the environment.

(a) State two ways that creating technology can have an effect on the environment. [2 marks]

(b) Explain two positive effects on the environment of using technology. [4 marks]

(c) Explain one negative effect on the environment of using technology. [2 marks]

2. An artist decides to use copyright to protect their latest artwork.

Explain what is meant by copyright and how it would protect the artist's work. [3 marks]

3. A city has an app that gives users information about all the different restaurants that are in the city. It can tell the user which restaurants are nearby and how to book a table with them.

Some people in the city have complained that the app uses location awareness to find out where in the city a user is, and feel this is an invasion of privacy.

Discuss the moral and ethical issues of using location awareness in this way. [6 marks]

TOPIC 7 UNDERSTANDING THE FUNCTIONALITY OF HTML

</CODE>
<html>
<head>
</head>
<body>
</body>
</html>
HTML CSS JAVA SCRIPT

Websites play an important role in modern life – they are used for many different tasks such as online banking, shopping, education and entertainment. In this topic you will learn how web pages are created using HTML, how the content of the web page can be structured and how features such as tables and forms are created. You will also learn how content such as audio and video files can be embedded in a web page and how links to other pages and external resources can be inserted into a page.

TOPIC 7 UNDERSTANDING THE FUNCTIONALITY OF HTML

LEARNING OBJECTIVES

By the end of this topic you should be able to understand:

- how HTML is used to structure web pages
- how to use semantic markup
- how to include images, tables, forms and media in a web page.

GETTING STARTED

You probably use a number of websites on a regular basis. What features do those websites have? What different types of media (e.g. images, videos, sound files) are included in those websites? Have you ever wondered how these features and media are included in the websites? Have you ever looked at or written any HTML? Perhaps you already know something about how HTML is written. For example, are you familiar with any of the tags used and what they do?

▶ A web designer exploring HTML templates

KEY TERMS

elements: components of an HTML document

tag: an item of markup that indicates the beginning and end of an HTML element

HTML DOCUMENT STRUCTURE

All websites are created using a markup language called HTML. HTML describes what text and other **elements** on the page (such as images and tables) should be included. HTML documents are sent from the web server to a browser, and the browser displays the web page based on the content of the HTML document. There are some minor differences in the way different browsers display certain HTML features, so, without checking the page with different browsers, you cannot aways be sure exactly how a web page will look to a user.

In most cases, HTML is used to create web page content but not formatting or layout. The formatting and layout of a web page is usually controlled using CSS (Cascading Style Sheets), which is contained in a separate file (although it can be included in the same file as the HTML if required). CSS is covered in the next topic.

HTML is created using a text file that can be edited using a text editor such as Windows Notepad. However, it is easier to write HTML if you use an editor designed for writing programming code, for example Notepad++, as it provides syntax highlighting and multiple open tabs. Notepad++ is free to download and use. HTML files should end with the .html extension, and once they have been created and saved they can be opened in any browser.

This is an example of a simple HTML document.

Head section with title

Body section

The <img> tag inserts an image. This is also explained later in the topic. See page 132.

```
 <!DOCTYPE HTML>
 <html lang="en">
    <head>
       <title>My first web page</title>
    </head>
    <body>
       <h1>Web page heading</h1>
       <p>
       Welcome to my web page
       </p>
       <img src="myImage.jpg" width="200" height="auto" alt="A dog">
       <p>
       Click <a href="linkPage.html">here</a> to visit my other page
       </p>
    </body>
 </html>
```

The <h1> tag creates a heading; the <p> tag below it creates a paragraph of text. These are explained later in the topic. See pages 134–136.

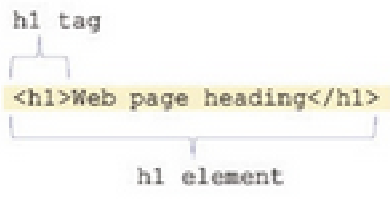
The <a> tag creates a link to another page. This is also explained later in the topic. See page 135.

HTML has a structure that defines the way information about the web page is provided. The main components of HTML are described in the following boxes.

Doctypes. All HTML documents should begin with a <!doctype> declaration. This tells the browser what type of document to expect. The correct doctype declaration for HTML 5 is:

```
      <!DOCTYPE html>
```

Pages written for earlier versions of HTML use a different doctype declaration.



Elements and tags. HTML elements combine together to make up the contents of the page and each element is defined by a **tag**. Elements are nested, with the root element defined by the <html> tag. Within the <html> element, the document is divided into the <head> element and the <body> element. Most elements end with a closing tag, which includes a forward slash and the element name, for example </html>. Omitting the end tag can lead to unpredictable results, so try not to forget it! We often refer to an element using the name of the tag that defines it.

KEY TERMS

attributes: define a property of an element

metadata: information about data; in HTML it provides information such as the character set used and the purpose of the web page

Attributes. These provide additional information about the element and are usually included in the start tag. Most **attributes** are in a 'name and value' pair. For example, this is an image element with a src attribute, which has a value of "myImage.jpg" to tell the img tag the source (file name) of the image to display. The alt attribute tells the browser what text to display if the image fails to load.

```
      <img src ="myImage.jpg" alt="A picture of a dog">
```

The language of the document. You should declare the language of the text on the web page. This is done in the <html> tag at the start of the document; for example:

```
      <html lang="en">
```

You can find the code for different languages by searching the internet for 'IANA language subtag registry'.

THE HEAD ELEMENT

The <head> element is used to provide details about various aspects of the web page, which is sometimes called **metadata**. It can contain the following information:

Document title. This is set using the <title> tag and is used in several ways. The title is shown in the title bar at the top of the browser window. It is the text that is shown in the user's favourites if they add the page to their favourites. The title will also appear in a search engine result for the page. Therefore, it is important that the title is accurate and reflects the content of the page.

Scripts. Program scripts, such as those written in JavaScript, can be placed in the head element; they start with a <script> tag and end with a </script> tag.

Styles. Styles are used to define the formatting and layout for elements on the page. CSS styles are covered in the next topic, but they can be defined here or linked from an external file, known as a style sheet.

Links to external files. Styles and scripts can be located in external files and link to the main HTML document using a <link> tag.

Meta. The <meta> tag is used to provide information about the page, such as the character set used, a description of the page, key words associated with the page and viewport settings. This data is used by the browser to display the page properly, and by search engines to help them understand what the page is about.

- **Character set.** It is recommended that the UTF-8 character set is used as this covers almost all the characters used in common languages.
- **Key words.** These should be added to reflect the main content and purpose of the page. Choosing the correct key words for a page is important because these are used by search engines. Key words can be single words or short phrases.
- **Description.** This should be a short description of the page. When a user sees the page listed in search engine results, the description appears in the results data. The description should therefore make the page appear interesting enough for users to click the link to view the page.
- **Refresh.** This option instructs the browser to refresh the page (download the latest copy from the server) after a certain time interval (in seconds). This could be used for a site that provides data such as stock market prices or exchange rates, which change very often.

The following code is an example of a header section of a page, including a title and meta tags.

```
<head>
    <title>Example page head section</title>
    <meta charset="UTF-8">
    <meta name="keywords" content="HTML, head section">
    <meta name="description" content="The HTML head section">
    <meta http-equiv="refresh" content="90">
</head>
```

Viewport. This should be included using a <meta> tag. The viewport is the size of the screen that the user is viewing the page on and will vary depending on whether they are using a mobile or a desktop device. The width attribute should be set to "device-width" so that the width of the page will use the width of the device screen, and the initial-scale attribute should be set to 1. The following code shows an example of this.

```
<meta name="viewport" content= "width=device-width, initial-scale=1.0">
```

WRITING WELL-ORGANISED HTML SYNTAX

A typical web page has a structure made up of headings, paragraphs and other elements such as lists and tables (see next section on Structural markup). For example, the <div> tag is used to create a division in the text of the page and group together related content.

In order to write clear, consistent and easy-to-read HTML, the following rules are recommended.

Most HTML syntax is not case sensitive, but it is considered good practice to write all element names, attributes and values in lower case.

As mentioned earlier, many HTML elements are nested. Indenting nested elements makes reading the HTML easier. You can see where each element starts and ends, and it is clear where elements have been nested.

Many attributes need to have their values in quotes. These can be double or single quotes, but for consistency double quotes should be used.

There are some HTML attributes that accept Boolean values (true, false). However, values of true and false should not be used. For example, the following:

```
<div hidden = "true">
```

is not valid; it should be:

```
<div hidden>
```

If the attribute is false it should be omitted. For example:

```
<div>
```

Some tags are known as self-closing in that they don't require a closing tag. The image tag <img> is an example of this. It is generally regarded as good practice to omit the closing forward slash, so:

```
<img src="myImage.jpg">
```

rather than:

```
<img src="myImage.jpg"/>
```

Other self-closing tags that you are likely to encounter include
 (new line), <hr> (horizontal rule), <meta> and <input> (to input data on a form).

USING GLOBAL ATTRIBUTES TO DEFINE ELEMENTS

Global attributes are those that can be applied to any element.

Class. The class attribute is used to specify a class name for an element. It is most commonly used with styles and is covered in the topic on CSS.

Id. The id attribute is used to define a unique ID for an HTML element. It is commonly used to identify an element for use within a script and so is covered in the topic on JavaScript. Later in this topic you will see how an id attribute can be used as bookmarks to link to a location on a page. An example of creating an ID for a paragraph is:

```
<p id="chapter4">
```

Style. Used to set a CSS style for an element.

Access key. Used to set a shortcut key to activate or give focus to the element. The value should be a single keyboard character. With most browsers the shortcut key must be used in combination with the alt key. An example is:

```
Click <a href="linkPage.html" accesskey="k">here</a>
```

Lang. Specifies the language for the element's text content. This is used if an element has text in a different language than the rest of the page. The default language for the page is set in the <html> tag as mentioned above.

Tabindex. Specifies the order of navigation through the page when using the tab key. It allows the developer to change the order in which elements on a page receive focus. For example:

```
<a href="news.html" tabindex="2">Latest news</a>
<a href="contacts.html" tabindex="1">Contact details</a>
<a href="home.html" tabindex="3">Home page</a>
```

In this example, the tab order for these three links has been changed using the tabindex attribute. The use of links is described in detail later in this topic. Users will expect a logical tab order on a page, so care should be taken and the order should not be changed without a good reason.

Data. Allows data to be stored within the web page. This data can then be accessed by JavaScript and used in a variety of ways. For example, the data could contain additional information about the element that could be displayed on request, by the user clicking a 'more information' button. The data attribute name must be followed by a dash and then the name the developer wants to give the data, which should be in lower case. For example:

```
<ul>
    <li data-details="A citrus fruit">Orange</li>
    <li data-details="Yellow in colour">Banana</li>
</ul>
```

In this example a data attribute called details (the li attribute) has been created within an unordered list (the <ul> tag). Unordered lists are described later in the topic.

Hidden. This attribute will mean that the element is not displayed. For example, it could be used to hide some website content, which is only displayed (by using JavaScript to remove the hidden attribute) when some other condition is met.

SKILLS

CREATIVITY, INTERPRETATION

ACTIVITY 1

1. What rules should you follow when writing HTML? Why is following the rules important?
2. What is metadata? How and why is metadata defined for a web page?
3. Create a simple web page with some information about something that interests you, or an information page about something you know about. Save the file on your computer with the .html extension.

   Be sure to include appropriate information in the head section of the HTML file you create.

   Open the web page in a browser and verify that it displays as expected.

▲ The link between metadata, the internet and users

STRUCTURAL MARKUP

A typical web page has a structure made up of headings, paragraphs and other elements such as lists and tables. Earlier versions of the HTML specification simply split elements, based on how they are used, into block and inline.

Block elements always start on a new line and take up the full width of the available browser window (although that can be changed with CSS). The most common block elements are <p> and <div>.

- The **paragraph tag** <p> is used to create a paragraph of text, preceded and followed by a new line. Note that inserting new lines (using the return key) into the text of the HTML document will not create new lines when viewed in a web browser as these are ignored. If you want a new line in the text on the page then you must either end the current paragraph (with </p>) and start a new one (with <p>), or use the break tag <br>. The browser will insert some space before and after a paragraph, but does not do this with a <br> tag. This makes the <br> tag useful where you don't want space between lines of text, for example in an address. The <br> tag is what is called an empty tag as it has no closing tag.
- The **<div> tag** is used to create a division in the text of the page and group together related content.

The <div> and <p> tags are known as **semantic elements**. They create a very similar effect when the web page is displayed in the browser, but their purpose is to identify different parts of the web page. If you have two paragraphs of text on the same topic, use the <p> tag to separate them. If you have two paragraphs of text on different topics, then use a <div> tag. There are several other semantic elements you can use to provide more options than just the <div> tag, and these are described later in this topic.

KEY TERMS

block elements: HTML elements that start on a new line and occupy the full width of the browser window

semantic elements: HTML elements that provide a definition of the purpose of the page content that the elements contain

KEY TERMS

content models: grouping of elements that have a particular purpose or behave in a similar way

inline elements: HTML elements that fit alongside the existing content (usually text) and don't need to start on a new line

Inline elements don't start on a new line (they can for example be 'inline' with the text on a page) and they only take up as much space as needed. Examples of inline elements include the <img> tag for inserting images, the <a> tag for inserting links and the <em> tag for emphasising text. All these tags are described in more detail later in this topic.

Block elements can contain inline elements, but inline elements cannot contain block level elements.

CONTENT MODELS

The current HTML specification divides the block and inline elements into several more **content models**. Each content model has a number of elements that apply to it.

Flow. This content model is the standard block level layout used within the body of an HTML document. All the other content models fit within this model.

Sectioning. This is a subset of flow content used to identify sections of related content on the page.

Heading. This is used to create a hierarchy of headings and subheadings for text. A series of 'h' tags, numbered h1 to h6, are used to provide headings, with h1 being the top level and h6 the bottom.

Phrasing. This is paragraph text (made up of sentences) and associated markup tags. Phrasing content is also a subset of flow content.

Embedded. This type of content embeds resources into the page. This includes images, audio and video.

Interactive. Used for content that the user can interact with such as buttons and links.

Examples of tags covered in this topic and the content model to which they belong are shown in Table 7.1. Note that some tags can belong to more than one content model.

TABLE 7.1: CONTENT MODELS AND EXAMPLES OF TAGS

Content model	Examples of tags that belong within the model
Flow	<div> – creates a division in the content
	<form> – creates an input form
	– ordered list
	– unordered list
	<p> – paragraph
	<table> – creates a table
Sectioning	<article> – semantic division of page content
	<aside> – semantic division of page content
	<section> – semantic division of page content
	<header> – semantic division of page content
	<footer> – semantic division of page content

(Continued)

TABLE 7.1: CONTENT MODELS AND EXAMPLES OF TAGS (*Continued*)

Content model	Examples of tags that belong within the model
Heading	<h1> to <h6> – creates different heading levels
Phasing	 – adds a line break <strong> – highlights text that is important <em> – provides text emphasis
Embedded	<audio> – includes an audio file in the page <iframe> – creates a frame that holds external interactive content <video> – includes a video file in the page <img> – includes an image file in the page
Interactive	<input> – creates various types of input controls on a form <select> – creates options for a drop-down box on a form <textarea> – creates a text input box on a form <button> – creates a clickable button

DEFINE THE STRUCTURE AND FORMATTING OF TEXT

Most web pages contain different sections to structure their content. Web pages that contain a lot of text content, such as blog posts, product reviews or academic articles, may have a number of different sections within them, including an introduction, sections of main content and summary, and so on. Traditionally, the different sections were defined using <div> tags, but using this method does not make the purpose of each section clear. HTML provides a number of semantic elements to provide a meaningful way of structuring content.

Articles. The <article> tag is used to create self-contained text content such as a blog post or a news article.

Sections. The <section> tag is used to create a section within the text content of the page, such as an introduction, summary or contact information.

Headings. The <header> tag creates introductory content such as the title for an article or section.

Paragraphs. As mentioned above, these are one or more sentences of text making up a paragraph.

Aside. The <aside> tag can be used to create an element for content that is less important than the main topics (articles, sections, etc.) of the page. It may contain information that is complementary to the main content but non-essential. It may be included in a sidebar on the page (which is styled with CSS).

Footer. The <footer> element will be at the bottom of the page and can contain information such as contact details, links, etc.

Thematic breaks. These are used where there is a change in the topic or story at the paragraph level. The <hr> tag is used to show a thematic break. This tag creates a horizontal rule (line) across the page, although this can be styled differently with CSS.

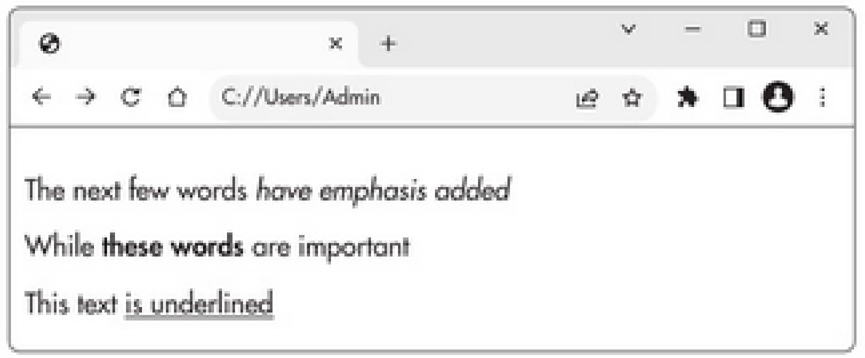
Emphasis. Within a paragraph of text you may want to emphasise certain words or phrases. This can be done using the tag, which is an inline element, and most browsers change the text following the tag to italics (although this can be modified with CSS). At the end of the text you want to emphasise the closing tag must be used. You can also set text to italics using the <i> tag, which must also be closed with an </i> tag.

Importance. Sometimes text within a paragraph is particularly important and you want to draw the reader's attention to it. For this you should use the <strong> tag. Typically, this element is displayed as bold (this also can be changed with CSS). In some cases you may want to indicate the importance of some text by using underline. This can be done using the <u> tag, but underline can be confused with a link (created with the <a> tag, which is also underlined).

The following code shows an example of the use of some of these tags.

```
        <p>
           The next few words <em> have emphasis added </em>
        </p>
        <p>
           While <strong>these words</strong> are important
        </p>
        <p>
           This text <u>is underlined</u>
        </p>
```

Figure 7.1 shows how this HTML will display in a browser.



The next few words *have emphasis added*

While **these words** are important

This text is underlined

▲ Figure 7.1 Use of tags to create emphasis and importance

Apart from making your HTML easier to read and understand for other web developers who might need to review or modify your code, there are other benefits to using semantic markup: accessibility and search engine optimisation. These are covered in detail in Topic 11.

Accessibility. For users with visual impairments it can be difficult to understand the content and purpose of the sections on a web page. Typically, these users will use screen reader software that can use the semantic markup to describe the page more accurately.

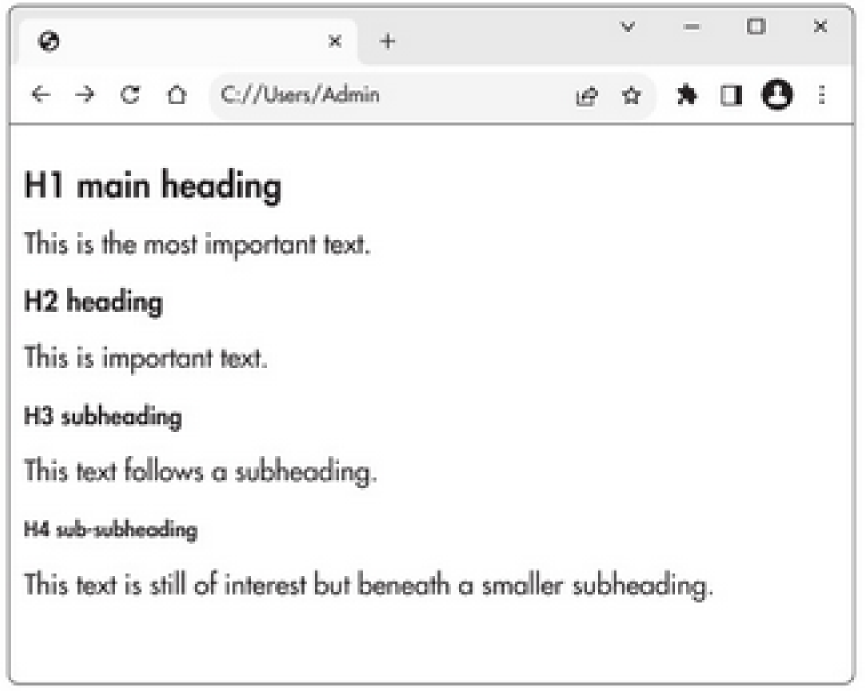
Search engine optimisation (SEO). Ensuring a web page ranks high in the listings when users search for a topic that relates to the page is very important for commercial websites. The teams that run these websites will spend a lot of time attempting to optimise the organisation's website to improve or maintain its search engine ranking. Using semantic content helps the web crawlers that search engines use understand the content of a page, so correctly used semantic markup will help improve a page's search engine performance.

Heading levels

The <h1> to <h6> tags are used to create a hierarchy of headings and subheadings within the text content of a page. The default text styling is set by the browser, but of course it can be modified with CSS. An example of the use of these headings is shown here:

```
   <!DOCTYPE HTML>
   <html lang="en">
      <head>
         <title>Heading levels</title>
      </head>
      <body>
         <h1>H1 main heading</h1>
            <p>This is the most important text.</p>
         <h2>H2 heading</h2>
            <p>This is important text.</p>
         <h3>H3 subheading</h3>
            <p>This text follows a subheading.</p>
         <h4>H4 sub-subheading</H4>
            <p>This text is still of interest but beneath a smaller
              subheading.</p>
      </body>
   </html>
```

Figure 7.2 shows this HTML displayed in a browser.



H1 main heading

This is the most important text.

H2 heading

This is important text.

H3 subheading

This text follows a subheading.

H4 sub-subheading

This text is still of interest but beneath a smaller subheading.

▲ Figure 7.2 Heading levels displayed in a browser

When using heading levels there are a few things to consider:

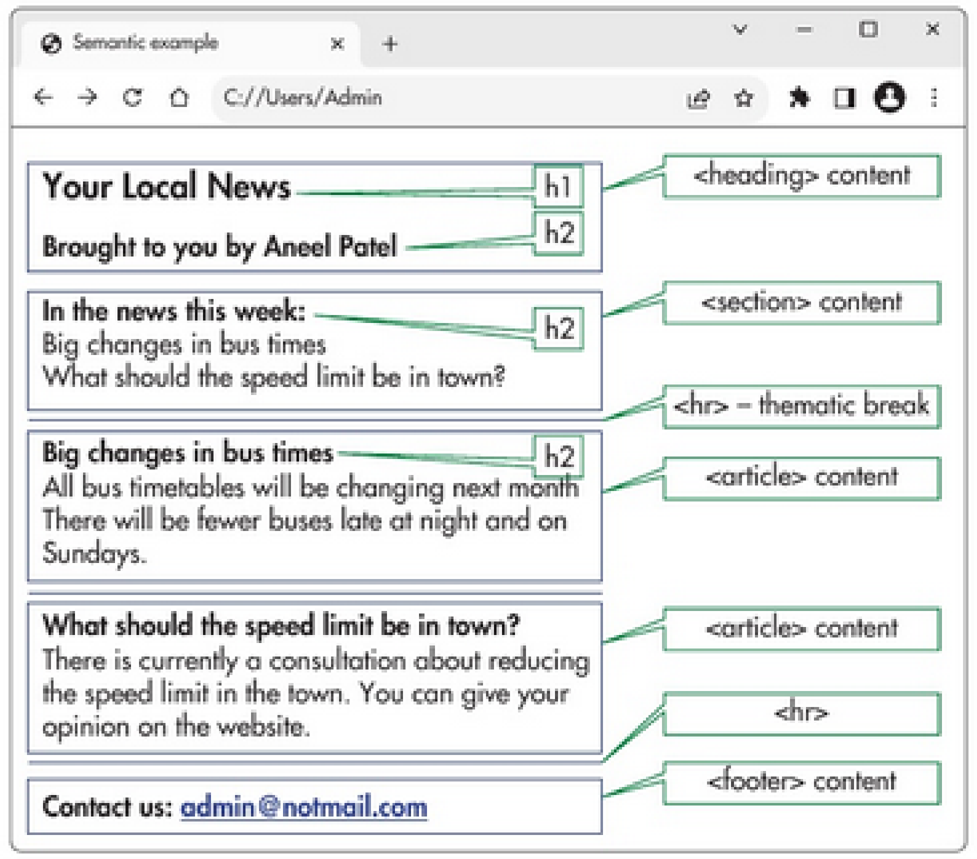
There should only be one h1 level per page and it should be used for the main topic of the page.

Don't skip any heading levels, so for example having only an h1 level and an h3 level on a page should be avoided (you would need an h2 level heading).

Order the heading levels by importance, with h1 being the most important, followed by h2, h3, and so on. The structure of the headings should follow a logical sequence, which will help search engines understand the text and support the page SEO. It will also help human users of the page understand the content.

Semantic markup example

Let's look at an example of how semantic markup could be used. Suppose you want to create a local news web page. Figure 7.3 shows an example of how a new page might be divided into different semantic sections.



Your Local News

Brought to you by Aneel Patel

In the news this week:
Big changes in bus times
What should the speed limit be in town?

Big changes in bus times
All bus timetables will be changing next month
There will be fewer buses late at night and on Sundays.

What should the speed limit be in town?
There is currently a consultation about reducing the speed limit in the town. You can give your opinion on the website.

Contact us: admin@notmail.com

▲ Figure 7.3 An example of semantic markup

The HTML for this page is:

```
   <!DOCTYPE HTML>
   <html lang="en">
   <head>
      <title>Semantic Example</title>
   </head>
   <body>
      <heading>
         <h1>Your Local News</h1>
         <h2>Brought to you by Aneel Patel</h2>
      </heading>
      <section>
         <h2>In the news this week:</h2>
         <p>
         <a href="#article1">Big changes in bus times</a><br>
         <a href="#article2">What should the speed limit be in town?</a>
         </p>
         <hr>
      </section>
      <article id="article1">
         <h2>Big changes in bus times</h2>
         <p>
         All bus timetables will be changing next month. There will be
           fewer buses late at night and on Sundays.
         </p>
         <hr>
      </article>
      <article id="article2">
         <h2>What should the speed limit be in town?</h2>
         <p>
         There is currently a consultation about reducing the speed
          limit in town. You can give your opinion on the website.
         </p>
         <hr>
      </article>
      <footer>
      <p><b>Contact us:</b> <a href = "mailto:
      admin@notmail.com">admin@notmail.com</a></p>
         <p>Contact us: admin@notmail.com</p>
      </footer>
   </body>
   </html>
```

Of course, without CSS to format and lay out the page it makes for a very plain web page, but the purpose is to demonstrate the use of structure and semantic elements within the text of a page. Formatting and layout are covered in the next topic on CSS.

Lists

As an alternative to paragraphs of text, lists of text can be created – they can be unordered, often called bulleted lists, or ordered (with numbers or letters). List elements are part of the flow content model.

An unordered list starts with the <ul> tag and each item in the list starts with the <li> tag. For example:

```
      <ul>
         <li>Apples</li>
         <li>Oranges</li>
         <li>Pears</li>
      </ul>
```

You can change the style of the bullets in a <ul> list using CSS.

Ordered lists start with the <ol> tag, with each item also starting with the <li> tag. By default, the list will be numbered. The type attribute allows you to choose a different type of list. The options are as follows.

- Type = “1” the list will be numbered (this is the default).
- Type = “A” the list will use uppercase letters.
- Type = “a” the list will use lowercase letters.
- Type = “I” the list will use uppercase roman numerals.
- Type = “i” the list will use lowercase roman numerals.

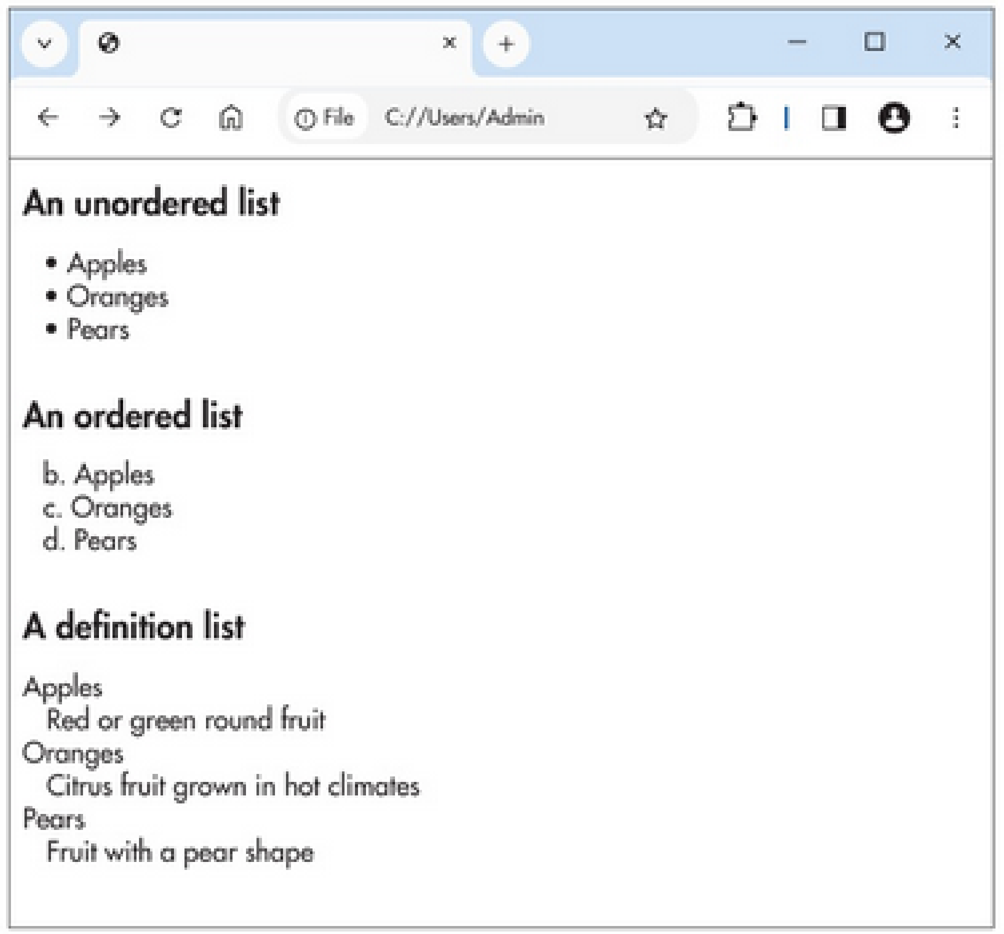
By default, a numbered list starts at 1 (or the equivalent) but you can change this with the start attribute. This example shows an ordered list with lowercase characters starting at “b”.

```
      <ol type="a" start="2">
         <li>Apples</li>
         <li>Oranges</li>
         <li>Pears</li>
      </ol>
```

You can also create definition lists with an indented description of each item. A definition list starts with the <dl> tag, the <dt> tag is used to start the term and the <dd> tag is used for the description of the term. For example:

```
      <dl>
         <dt>Apples</dt>
            <dd>Red or green round fruit</dd>
         <dt>Oranges</dt>
            <dd>Citrus fruit grown in hot climates</dd>
         <dt>Pears</dt>
            <dd>Fruit with a pear shape</dd>
      </dl>
```

Figure 7.4 shows how the three types of lists, when put together in a single HTML document, would display in a browser.



An unordered list

- Apples
- Oranges
- Pears

An ordered list

b. Apples
c. Oranges
d. Pears

A definition list

Apples
  Red or green round fruit
Oranges
  Citrus fruit grown in hot climates
Pears
  Fruit with a pear shape

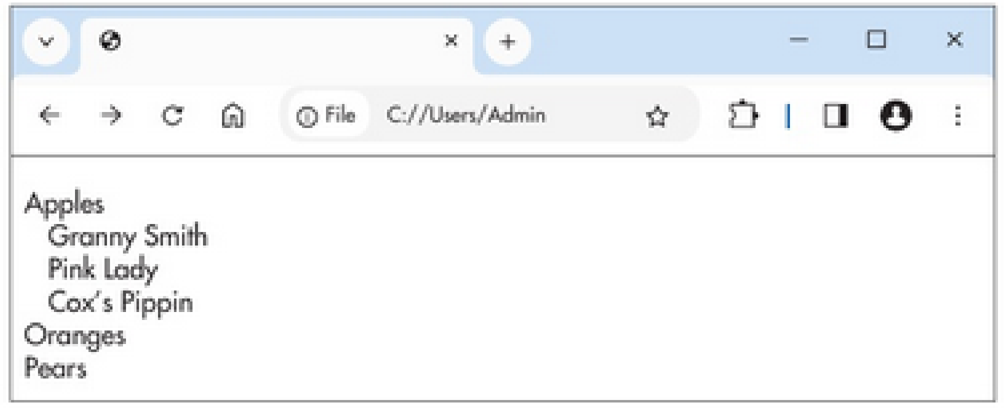
▲ Figure 7.4 List elements

Nesting lists. Lists can be nested within other lists. Figure 7.5 shows the HTML for a nested list and how it would be displayed in a browser.

```
   <ol type="a">
      <li>Apples
         <ol type="i">
         <li>Granny Smith</li>
         <li>Pink Lady</li>
         <li>Cox’s Pippin</li>
         </ol>
      </li>
      <li>Oranges</li>
      <li>Pears</li>
   </ol>
```

Inner list, nested inside the outer list.

Outer list.



Apples
- Granny Smith
- Pink Lady
- Cox's Pippin

Oranges
Pears

▲ Figure 7.5 A nested list

Links

Links (known as hyperlinks) can be inserted in a page using the <a> tag. The href attribute identifies the location of the link. The link itself can be text (add the tag after the link text) or, if you put an <img> tag inside the <a> tag, an image. The browser will underline a text link and show it in a different colour depending on whether the link has been visited or not. By default, unvisited links will be blue and a link that has been visited will be purple. As you click on a link it will turn red. Links can point to various resources.

Internal. You can create a link to another page on the same website simply by setting the value of the href attribute to the file name of the HTML document. If the file is in a different folder then you will need to include the relative path, such as "/main/home.html".

External. Links to external web pages can be made by setting the href attribute to the URL of the page you want to link to. Note that you need to include the full URL including the http:// part.

Email. By using the mailto: value with the href attribute, you can provide a link that opens the user's default email program with the email address that you specify.

This code is an example of these three methods of creating links.

```
<h1>Web page heading</h1>
<p>Welcome to my web page</p>
<p>
Click <a href="https://www.Pearson.com">here</a> to see latest news
</p>
<p>
Click <a href="linkPage.html">here </a> to view my other page
</p>
<p>
Click <a href = "mailto:me@notmail.com">here</a> to email me
</p>
```

External link

Internal link

Email link

As we've already mentioned, the link doesn't have to be text; you can use an image as a link by including the <img> tag inside the <a> tag. You can also add a title attribute to the <a> tag, which will display the title value as a tool tip when you move your mouse over the link. This code example shows an image as a link with the title attribute.

```
<a href="home.html" title="go to home page"><img src="myImage.jpg"
  width="100" height="auto" alt="A home icon"></a>
```

The target attribute. By default, the link will open in the same browser window or tab, but you can change that with the target attribute. You can set the target attribute to the following:

- **_self** (the default) opens the link in the same window or browser tab
- **_blank** opens the link in a new window or tab.

Link within the same page. The link can point to another location on the same page (sometimes called a bookmark or anchor). You need to precede the link name with a hash symbol and use an id attribute to identify the place on the page where the link should go. For example, you could create a bookmark for a paragraph on the page like this:

```
<p id="chapter4">
```

And then create a link to that bookmark like this:

```
<a href="#chapter4">Go to chapter 4</a>
```

You can also link to a specific place on another internal page by preceding the name of the id with the name of the HTML file. For example:

```
<a href="semantic.html#article2">Go to article 2</a>
```

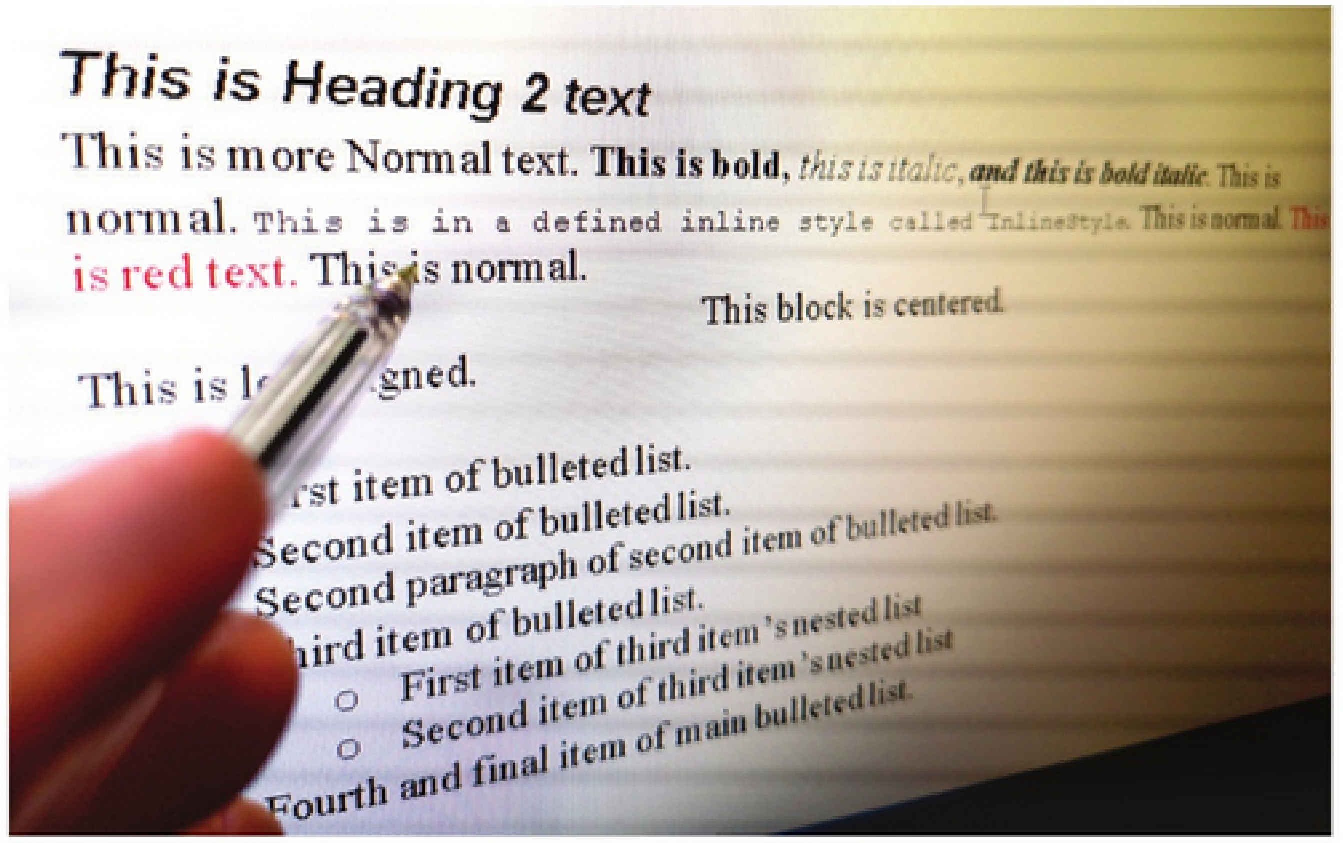
SKILLS
CRITICAL THINKING, ANALYSIS

ACTIVITY 2

1. What is the difference between an ordered list, an unordered list and a definition list?
2. What is semantic markup? Why should you use it?
3. What is the difference between an internal and external link?
4. Create a web page that includes the list below.

   Remember to save your file with the .html extension.
   - **a.** Unit 1
   - **b.** Unit 2 – Web development
     - Topic 7: Understanding the functionality of HTML
       - **i.** Document structure
       - **ii.** Structural markup
       - **iii.** Page components
     - Topic 8: Understanding the functionality of CSS
     - Topic 9: Understanding the functions of JavaScript
     - Topic 10: Designing web pages
     - Topic 11: The semantic web
   - **c.** Unit 3

5. Open the HTML file you created in a web browser and confirm that it works as expected. Make any changes until your web page looks exactly as you want.



This is Heading 2 text

This is more Normal text. **This is bold,** *this is italic,* ***and this is bold italic.*** This is normal. `This is in a defined inline style called InlineStyle.` This is normal. This is red text. This is normal.

This block is centered.

This is l gned.

rst item of bulleted list.

Second item of bulleted list.

Second paragraph of second item of bulleted list.

hird item of bulleted list.

○ First item of third item's nested list

○ Second item of third item's nested list

Fourth and final item of main bulleted list.

▲ Exploring HTML headings and text formatting

PAGE COMPONENTS

There are many other components you can add to a web page. In this section you will look in detail at how images are inserted, how to create tables and forms and how to embed audio and video files.

ADDING IMAGES TO WEB PAGES

Images are linked to a web page with the <img> tag, which has two attributes that need to be included. The src attribute identifies the file name of the image. If the image is in a different folder from the HTML document then you need to specify the path to the folder in the src attribute. You also need to include the alt attribute, which is described later in this section. For example: <img src ="images/myImage.jpg" alt="A picture of a dog">

File formats. Image files need to be in a file format that is supported by most browsers. For photographic images, JPG is the most suitable choice, while logos and diagrams can use the PNG format. The file formats supported by all common browsers are listed in Table 7.2.

TABLE 7.2: GRAPHICS FORMATS

File format	File extension	Used for
Animated Portable Network graphics	.apng	Animated graphics. APNG is an extension of the PNG format and supports more colours than GIF
Graphics Interchange Format	.gif	Animated graphics
Microsoft Icon	.ico	Small icons
Joint Photographic Expert Group	.jpg, .jpeg	Photographic images
Portable Network Graphics	.png	Diagrams, charts, illustrations
Scalable Vector Graphics	.svg	Vector graphics such as diagrams

Image size. If you don't specify the image size then it will be displayed actual size. Therefore, it is wise to set the width and height attributes. These attributes are measured in pixels.

Resolution. The resolution of the image does not change when you resize it, so if you set the size attributes much larger than the original image the result will be pixelated.

Proportions. You need to set the height and width of an image in proportion to the original image size, otherwise the result will look distorted. To avoid this problem you can set either the height or width of the image to 'auto' so that the image will be scaled in proportion.

Position. Images behave as inline elements so will appear alongside any text unless you use a <div> tag to place them on a separate line. Precise positioning of images can be achieved using CSS.

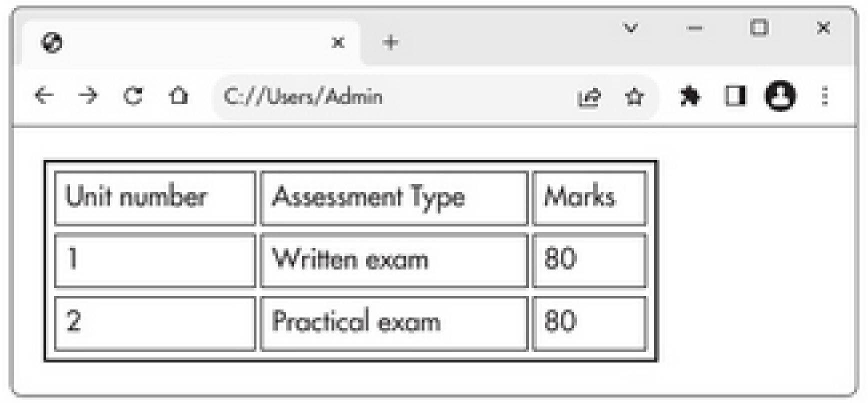
Alt attribute. All image tags should include an alt attribute. This is used to describe the image in a situation where the user cannot view it. This might be because of a slow connection or because the user is visually impaired and is using a screen reader. The screen reader will read the alt tag to the user so they can understand what the image shows.

PRESENTING INFORMATION IN A TABLE

A table is a useful way of presenting text and numerical data that fits into a row and column layout. Tables can be created using the <table> tag. Rows are added to the table using the <tr> tag, and data cells within the rows are added using the <td> tag. Here is a simple table example:

```
<table>
    <tr>
        <td>Unit number</td>
        <td>Assessment Type</td>
        <td>Marks</td>
    </tr>
    <tr>
        <td>1</td>
        <td>Written exam</td>
        <td>80</td>
    </tr>
    <tr>
        <td>2</td>
        <td>Practical exam</td>
        <td>80</td>
    </tr>
</table>
```

Figure 7.6 shows how this table looks when displayed in a browser.



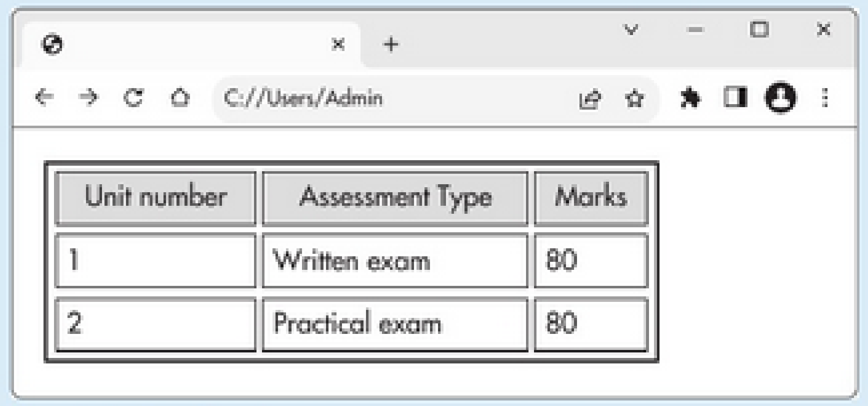
Unit number	Assessment Type	Marks
1	Written exam	80
2	Practical exam	80

▲ Figure 7.6 A basic table

You can add **heading rows** using the <th> tag rather than <td>; heading rows are bold, centred and have a grey fill. This feature is commonly used for the top row of the table that contains the titles. For example, the top row of the table shown in Figure 7.6 could be modified to use heading rows like this:

```
<tr>
    <th>Unit number</th>
    <th>Assessment Type</th>
    <th>Marks</th>
</tr>
```

When displayed in a browser, the table will now look like Figure 7.7.



Unit number	Assessment Type	Marks
1	Written exam	80
2	Practical exam	80

▲ Figure 7.7 A table with a header row

You can create tables with header, body and footer sections. These allow large tables to have the body content scrolled while the header and footer stay in place, so that titles at the top of the table (headers) don't disappear when you scroll down the table. The header section is created with the <thead> tag, the body section with the <tbody> tag and the footer section with the <tfoot> tag. The following code is an example of how a section of a table can be identified as the header using the <thead> tag.

```
<thead>
    <tr>
        <th>Unit number</th>
        <th>Assessment Type</th>
        <th>Marks</th>
    </tr>
</thead>
```

Combine multiple cells. You can combine multiple cells within a table using the colspan or rowspan attributes of the <td> tag. The value the attribute is set to is the number of rows or columns that the cell should combine. For example, using the table we have just created the last column could have the second and third rows combined by using the attribute rowspan=2 so they span over two rows, like this:

```
<table>
    <tr>
        <th>Unit number</th>
        <th>Assessment Type</th>
        <th>Marks</th>
    </tr>
    <tr>
        <td>1</td>
        <td>Written exam</td>
        <td rowspan="2">All units 80 marks</td>
```

This data cell uses the rowspan attribute so it will occupy two rows in the table.

```
    </tr>
    <tr>
        <td>2</td>
        <td>Practical exam</td>
    </tr>
</table>
```

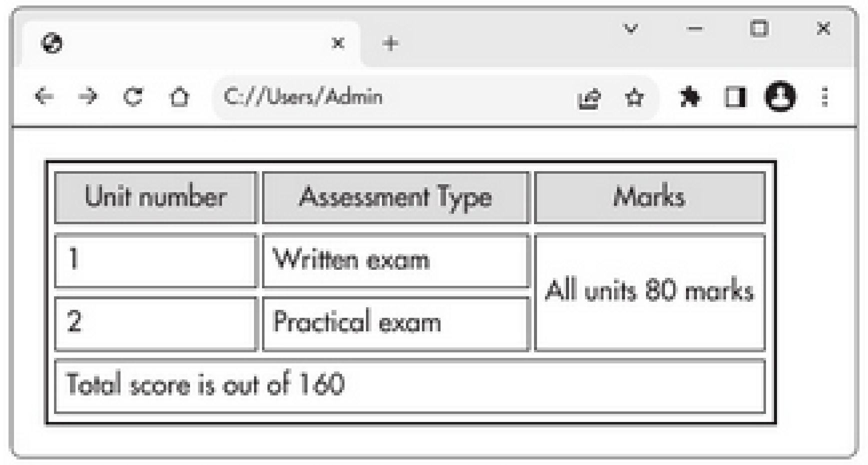
This row now only needs two data cells as the third one is taken up by the cell using the rowspan.

The following code is an example of how a section of a table can be identified as the footer using the <tfoot> tag.

```
<tfoot>
    <tr>
    <td colspan="3">Total score is out of 160</td>
    </tr>
</tfoot>
```

This data cell uses the colspan attribute so it will occupy three columns in the table.

Figure 7.8 shows this table displayed in a browser.



Unit number	Assessment Type	Marks
1	Written exam	All units 80 marks
2	Practical exam	
Total score is out of 160		

▲ Figure 7.8 A table with combined cells

The table borders shown in all the tables above are the default ones. You can modify the thickness and style of the borders using CSS.

CREATING A FORM ON A WEB PAGE

Forms allow websites to collect data from users, such as allowing them to register to receive updates from an organisation. Forms are created with the <form> tag, and there are a variety of input controls that can be added to a form using the <input> tag. Here are the most commonly used ones:

Text. To create a text input box the tag <input type="text"> is used. A label should be added to the input box, so the user knows what to enter, and is added using the <label> tag.

Radio buttons. The <input type="radio"> tag creates a set of buttons a user can select, of which only one can be selected.

Checkbox. The <input type="checkbox"> tag creates a set of check boxes for the user to select; multiple check boxes can be selected (unlike radio buttons).

Submit button. This is added using the <input type="submit"> tag. The button is used to submit the form to the form handler on the server. This is explained in the section on how forms work.

KEY TERMS

checkbox: a form control where any of the options can be selected

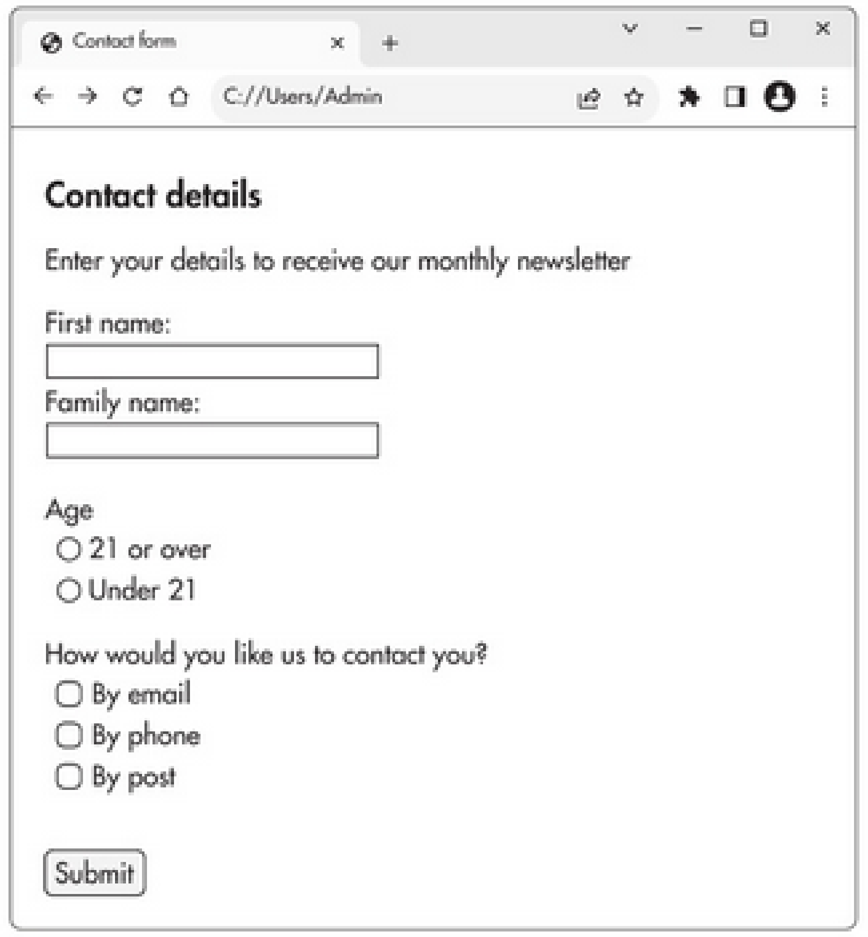
forms: a method of user input

radio buttons: a form control where only one of the group of options can be selected

This code shows an example of a form using these controls.

```
<!DOCTYPE HTML>
<html lang="en">
<head>
   <title>Contact form</title>
</head>
<body>
   <h2>Contact details</h2>
   <p>Enter your details to receive our monthly newsletter</p>
   <form>
      <label>First name:</label><br><input type="text"><br>
      <label>Family name:</label><br><input type="text"><br><br>
      <label>Age</label><br>
      <input type="radio" name="age"><label>21 or over</label><br>
      <label>Under 21</label><br><br>
      <label>How would you like us to contact you?</label><br>
      <input type="checkbox"><label>By email</label><br>
      <input type="checkbox"><label>By phone</label><br>
      <input type="checkbox"><label>By post</label><br><br>
      <input type="submit">
   </form>
</body>
</html>
```

This example form uses <br> tags to insert new lines into the page and provide spaces between form controls. Figure 7.9 shows how this form would appear in a browser.



Contact details

Enter your details to receive our monthly newsletter

First name:

Family name:

Age

○ 21 or over

○ Under 21

How would you like us to contact you?

☐ By email

☐ By phone

☐ By post

Submit

▲ Figure 7.9 An example contact form

Table 7.3 shows other input types that you can use on a form.

TABLE 7.3: FORM INPUT TYPES

Input type	Use
<input type="button">	Creates a button. The text on the button can be set with the value attribute. JavaScript code is needed to make the button carry out an action.
<input type="date">	Will display a date picker when clicked, ensuring only a valid date can be entered. (There are also day, month and week input types that show a date picker and allow a month or week and year to be selected.)
<input type="email">	Creates a text input box that is validated as an email (i.e. containing an @ symbol and at least one full stop).
<input type="password">	Creates a text input box but the characters entered in the box are hidden.
<input type="reset">	Creates a reset button, which when clicked will clear the contents of the form.
<input type="number">	Creates an input box that will only accept numbers. You can also set upper and lower limits using max and min attributes.

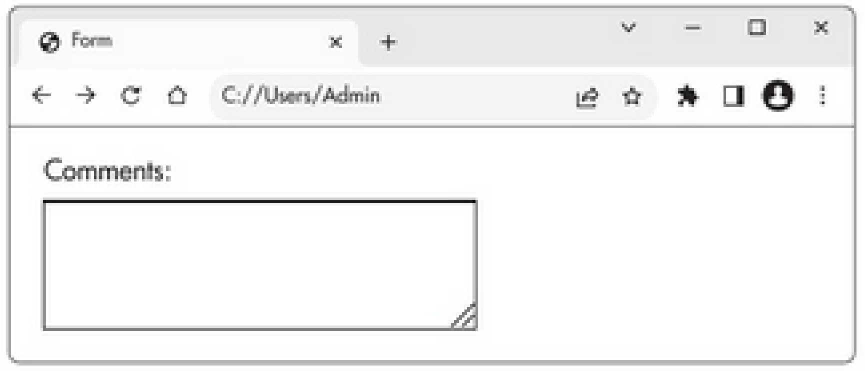
Here are some other types of controls you can add to a form:

Text area. The <textarea> tag creates a multi-line text box. You can state the number of columns and lines that are included in the text box. For example:

```
<label>Comments: <br></label><textarea rows="3"
cols="20"></textarea>
```

In this example there is nothing between the <textarea> opening tag and the closing </textarea> tag. Anything between the tags will appear in the text area box.

This will display in the browser as shown in Figure 7.10.



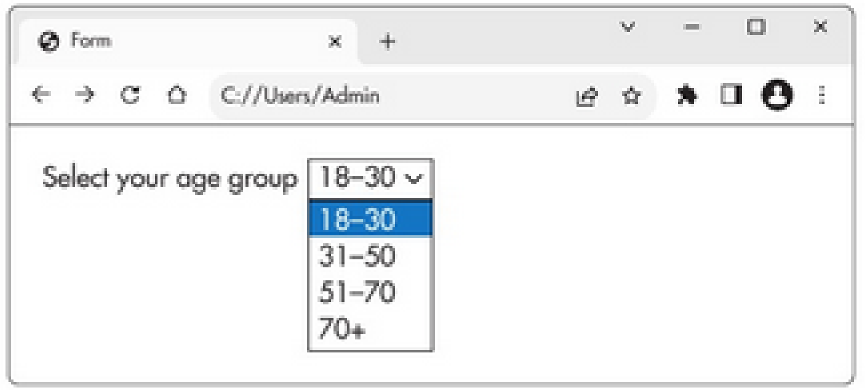
Comments:

▲ Figure 7.10 A text area input box

Drop-down list. To create a list that drops down when clicked, so the user can select one of the options listed, the <select> tag is used, along with a number of <option> tags to create the items listed in the drop-down box. The size attribute can be added to the select tag to define how many items show in the box before it is dropped down. The following code is an example of the use of the <select> tag:

```
<label>Select your age group</label>
<select name="agegroup">
   <option value="18-30">18-30</option>
   <option value="31-50">31-50</option>
   <option value="51-70">51-70</option>
   <option value="70+">70+</option>
</select>
```

This will display in the browser as shown in Figure 7.11 (with the box dropped down):



Select your age group 18–30

18–30
31–50
51–70
70+

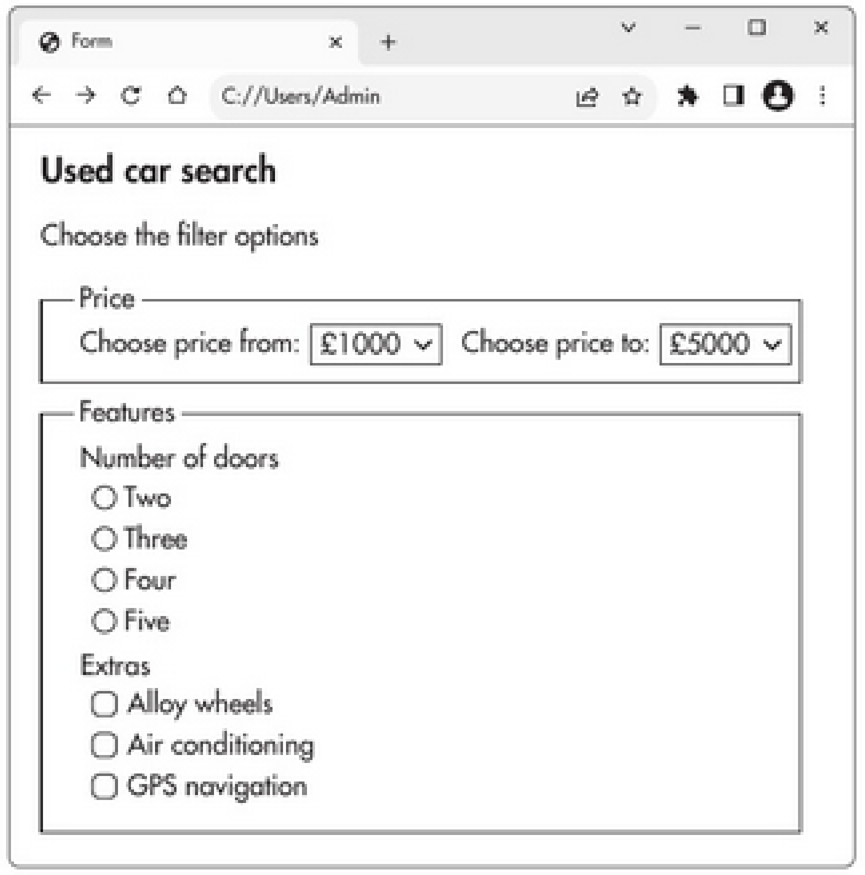
▲ Figure 7.11 A form drop-down box

By default, only one item in the list can be selected. Adding the 'multiple' attribute will allow the user to select more than one option.

Grouping form elements. Form elements can be grouped together using the <fieldset> attribute and a legend (title) added to the field set with the <legend> tag. This code is an example of a form with grouped sections using the <fieldset> tag.

```
   <h1>Used car search</h1>
   <p>
      Choose the filter options
   </p>
   <form>
      <fieldset> <legend>Price</legend>
         <label>Choose price from:</label>
         <select name="Price From">
            <option value="1">£1000</option>
            <option value="2">£3000</option>
            <option value="3">£5000</option>
         </select>
         <label>Choose price to:</label>
         <select name="Price To">
            <option value="1">£5000</option>
            <option value="2">£8000</option>
            <option value="3">£10000</option>
         </select>
      </fieldset>
      <fieldset><legend>Features</legend>
         <label>Number of doors</label><br>
            <input type="radio" name="doors2"><label>Two</label><br>
            <input type="radio" name="doors3"><label>Three</label><br>
            <input type="radio" name="doors4"><label>Four</label><br>
            <input type="radio" name="doors5"><label>Five</label><br>
         <label>Extras</label><br>
            <input type="checkbox"><label>Alloy wheels</label><br>
            <input type="checkbox"><label>Air conditioning</label><br>
            <input type="checkbox"><label>GPS navigation</label><br>
      </fieldset>
      </form>
```

Figure 7.12 shows how this form will appear in a browser.



Used car search

Choose the filter options

Price
Choose price from: £1000 Choose price to: £5000

Features
Number of doors
○ Two
○ Three
○ Four
○ Five
Extras
☐ Alloy wheels
☐ Air conditioning
☐ GPS navigation

▲ Figure 7.12 A form with groupings

HOW WEB FORMS WORK

The forms you have looked at so far can be displayed and filled in, but there is no mechanism to process the data that has been entered. The developer needs to provide a script to process the form data, which could be written in PHP, for example. To make this work, additional information needs to be added to the <form> tag:

Action attribute. This needs to be set to the name of the web page or script that will process the form data.

Method attribute. This is the method that is used to send the data to the processing page or script. There are two methods that can be used:

GET. This method appends the form data to the URL of the processing page or script in a series of ID and value pairs. This data will be seen in the address bar when the processing page loads. This is one reason why any sensitive data should not be sent this way.

POST. This method, which is generally preferred, appends the form data to the HTTP (Hyper Text Transmission Protocol) request itself so it is not visible, but it can be accessed by the processing script.

This code is an example of a <form> tag with the action and method attributes.

```
   <form action="process.php" method="POST">
```

Form validation. As with any input data that will be processed by a program, validating the input is important to ensure the program can handle the data and it is reasonable for the file in question. There are two main ways data on the form can be validated:

Client-side validation. This is where the validation is done within the form page itself. There are some ways that the input element can do some validation, such as using the date types so that invalid data is not possible. Numeric input types can also prevent invalid text entries and the min and max attributes can be used to limit the range of values allowed. The other way validation can be done on the page is using JavaScript.

Server-side validation. This should be avoided where possible since it places extra processing load on the web server, but some validation can only be done on the server. An example would be validation of email addresses, where an email needs to be sent to the email address provided on the form for confirmation.

ADDING AUDIO AND VIDEO

You can embed assets such as video and audio into web pages. There are many different media formats that are used, not all of which will work in all browsers.

Video. For display on web pages the MP4, Web4 and Ogg formats are recommended. Formats such as QuickTime and AVI won't play on some browsers. To insert a video into a web page, use the <video> tag. This code is an example of how to use the video tag.

```
      <video width="300" height="200" controls>
         <source src="myVideo.mp4" type="video/mp4">
      </video>
```

Setting the height and width of the video window is recommended to ensure smooth playback. The controls attribute will display the play, pause buttons and volume control. The appearance of the controls can be customised with JavaScript and CSS. This is covered in Topic 9 (JavaScript).

Several additional attributes can be included in the <video> tag:

- **Autoplay** – sets the video to play automatically without the user having to press the play control (not supported on all browsers)
- **Loop** – sets the video to play in a continuous loop
- **Muted** – sets the audio output to be muted
- **Poster** – sets the file name or URL of an image to be displayed in the video window until the video has downloaded (and until the user presses the play button).

Audio. Audio files can be inserted into a web page in a similar way to video. MP3 and WAV formats are widely supported by browsers. For example:

```
      <audio controls>
         <source src="myAudio.mp3" type="audio/mpeg">
      </audio>
```

As with the video tag, the control attribute displays the player controls. Omitting the attribute will mean the controls do not appear. The appearance of the controls can be customised with JavaScript and CSS.

Several additional attributes can be included in the <audio> tag:

- **Autoplay** – sets the audio to play automatically without the user having to press the play control
- **Loop** – sets the audio to play in a continuous loop
- **Muted** – sets the audio output to be muted.

Multiple sources. One method you can use to avoid possible incompatibility between the audio/video file format and the browser that the user has, is to list multiple source files. For example:

```
      <audio controls>
         <source src="myAudio.mp3" type="audio/mpeg">
         <source src="myAudio.ogg" type="audio/ogg">
         Your browser does not support audio
      </audio>
```

The browser will use the first format it is able to play. The text 'Your browser does not support audio' is only displayed if none of the formats are supported.

Table 7.4 lists the audio and video formats supported by most browsers.

TABLE 7.4: SUPPORTED AUDIO AND VIDEO FORMATS

Format	Type	Comments
MP3	Audio	
WAV	Audio	WAV files are uncompressed so files can be large
OGG	Audio and video	Not supported by Safari
MP4	Video	
WebM	Video	

USING INLINE FRAMES

An inline frame is an effective way to include content from other websites, such as YouTube, in a web page. It is particularly useful with video as it avoids any problems with the browser used being incompatible with the format of the video. An example of an inline frame that will play a YouTube video is:

```
   <iframe width="400" height="310"
   src="https://www.youtube.com/embed/abc1234">
   </iframe>
```

The width and height attributes are used to set the size of the frame in which the content appears. To make this work you need to insert the ID of the YouTube video, which is placed at the end of the URL (shown as abc1234 here). To find the ID, right click on the video on You Tube and choose 'Stats for nerds', and the ID is shown at the top of the pop-up that appears.

Inline frames can also be used to display any type of web page within the frame. It is recommended that the title attribute is included as it will enable screen readers to read out the title or content of the frame (like the alt attribute used with images).

SKILLS

INTERPRETATION, CREATIVITY, ADAPTIVE LEARNING

ACTIVITY 3

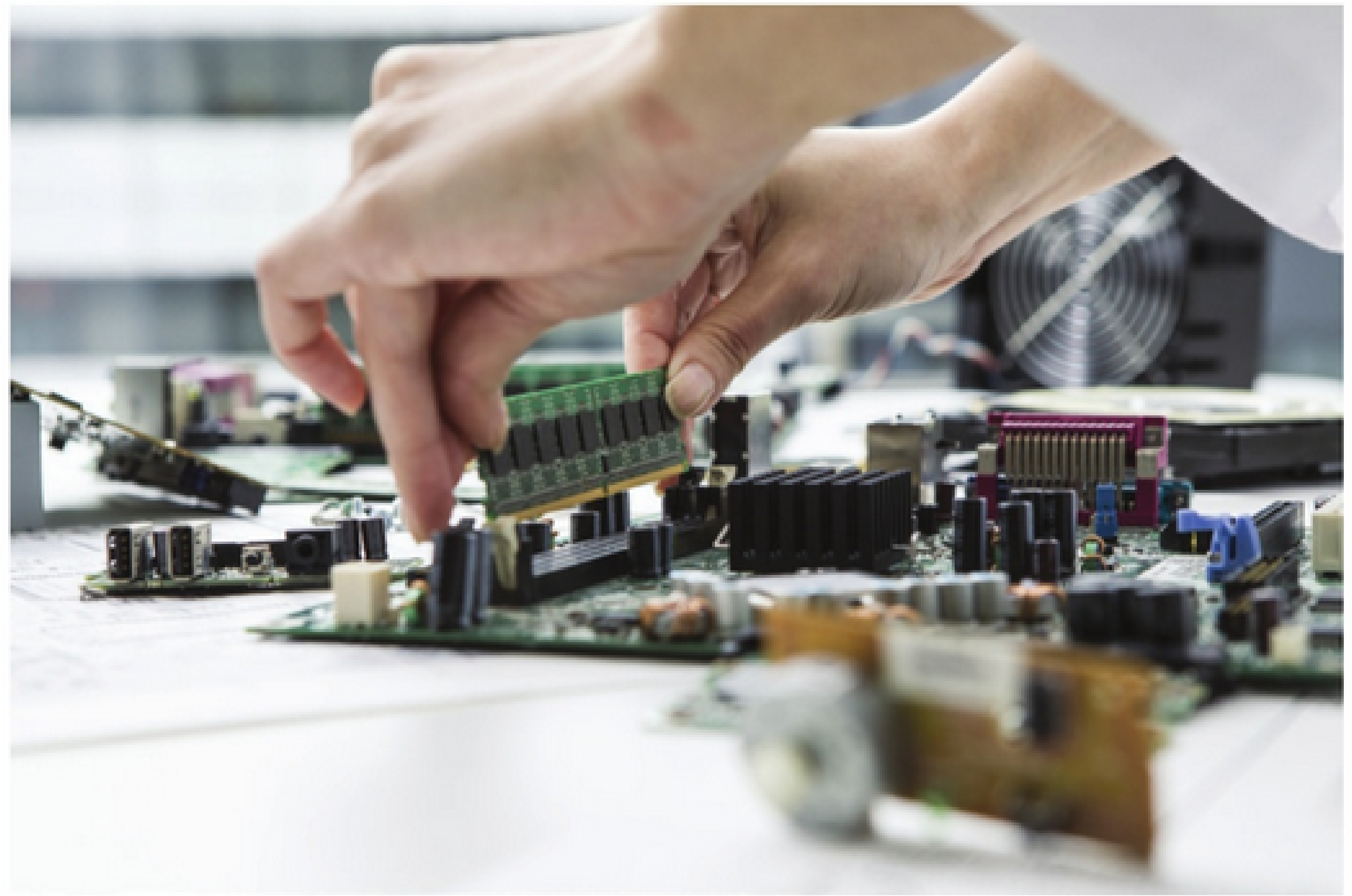
1. Create a web page with a table that provides details of the attributes of common HTML tags. It should look something like this:

Tag	Attributes	Values
<a> Used to create a link	href – used to specify the URL of the link	Can be an internal file or external URL
	target – used to specify if the link should open in the same window or a new window or tab	_self – open in the same window (default) _blank – open in new window or table
	title – used to provide a tool tip	
<img> Used to embed an image	src – the source of the image file	Can be a variety of image file formats
	alt – provides alternate text to describe the image	
	height and width – sizes the image in pixels	One can be set to "auto" to keep the image in proportion

The completed page can provide you with revision notes for your exam.

Open the created web page in a browser and check that it displays as expected.

2. Create a web page containing a form for a customer to select components to build their own computer. This form should allow users to select all the different components that you learned about in Topic 1. The form should include as many different form controls as possible including radio buttons, check boxes, drop-down options and a submit button. It should also include <fieldset> groupings.



▲ Building a computer using standard components

CHECKPOINT

1. How can you create a link to another website from a web page?
2. List two examples of good practice when writing HTML.
3. What is the purpose of the <p> tag?
4. Name two different content models.
5. What graphics file format is most commonly used for photographic images?
6. In an ordered list, how can you number the items with lowercase roman numerals?
7. What tag is used to create a thematic break?
8. What will be the effect of applying the <strong> tag to some text?
9. Where and why should the alt attribute be used?
10. What video formats are supported by all browsers?

THINKING LIKE AN IT SPECIALIST

An IT specialist needs to be able to use a range of software tools for different purposes. In this topic we have been using Notepad++ to directly edit HTML code, but there are typically a number of different software tools used to develop web pages. These include tools such as web page editing applications and graphics editors. Are there other tools you are familiar with or have heard of that can be used in the web page development process? What are the benefits and drawbacks of those tools when compared with Notepad++?

EXAM TIP

The exam paper usually has the first few questions about HTML. You may be asked to look at some HTML and identify any errors. You might also be asked to write a small amount of HTML code to implement a particular feature, such as creating an iframe or hyperlink.

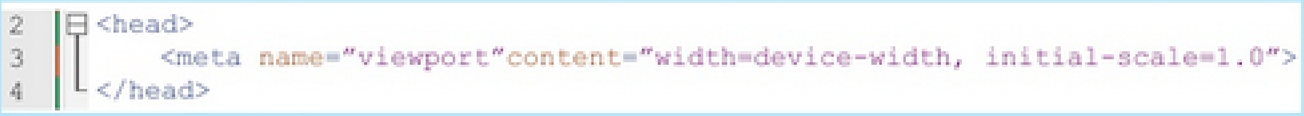
SKILLS

INTERPRETATION, PROBLEM SOLVING, CRITICAL THINKING

EXAM PRACTICE

1. Web pages can use meta tags.

The HTML code in Figure 7.13 contains a meta tag.



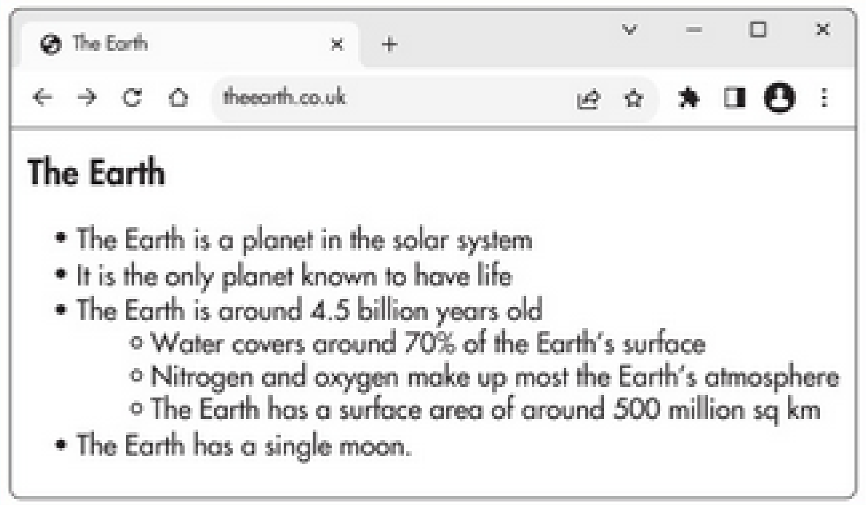
```
2  <head>
3      <meta name="viewport"content="width=device-width, initial-scale=1.0">
4  </head>
```

▲ Figure 7.13 HTML code

Explain the purpose of the HTML code on line 3. [3 marks]

2. Figure 7.14 shows how a list of facts about the Earth should be displayed in a browser. The HTML code should:

- use a bullet point for each sentence
- nest the three bullet points starting 'Water covers…' under the main bullet list.

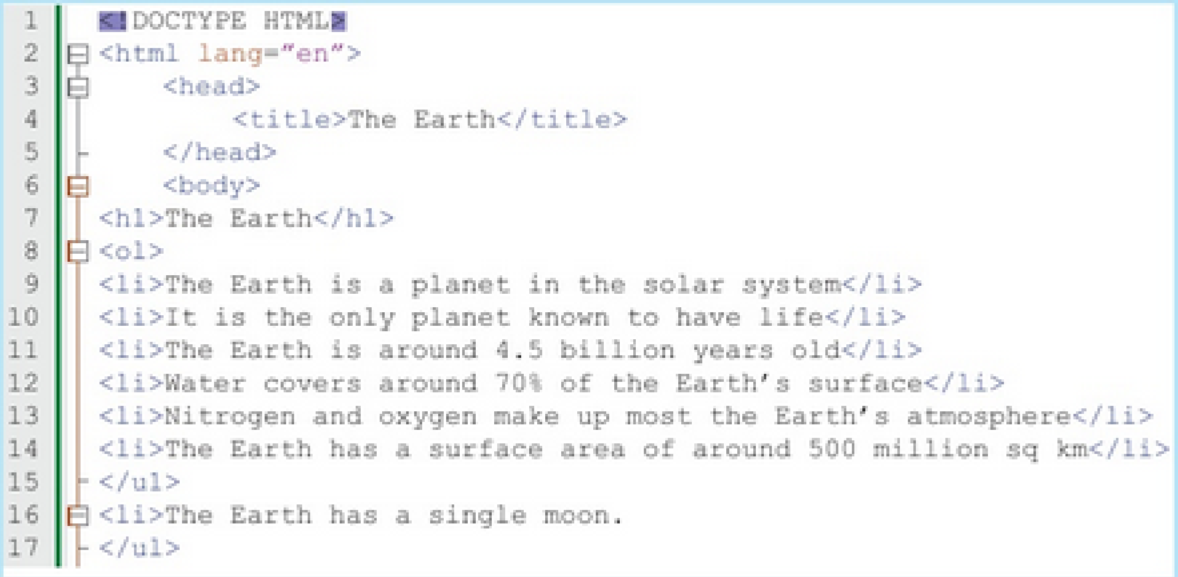


The Earth

- The Earth is a planet in the solar system
- It is the only planet known to have life
- The Earth is around 4.5 billion years old
    - Water covers around 70% of the Earth's surface
    - Nitrogen and oxygen make up most the Earth's atmosphere
    - The Earth has a surface area of around 500 million sq km
- The Earth has a single moon.

▲ Figure 7.14 Web page about the Earth

The page does not display as intended because there are **two** errors in the HTML code shown in Figure 7.15.



```
<!DOCTYPE HTML>
<html lang="en">
    <head>
        <title>The Earth</title>
    </head>
    <body>
<h1>The Earth</h1>
<ol>
<li>The Earth is a planet in the solar system</li>
<li>It is the only planet known to have life</li>
<li>The Earth is around 4.5 billion years old</li>
<li>Water covers around 70% of the Earth’s surface</li>
<li>Nitrogen and oxygen make up most the Earth’s atmosphere</li>
<li>The Earth has a surface area of around 500 million sq km</li>
</ul>
<li>The Earth has a single moon.
</ul>
```

▲ Figure 7.15 Code for making the web page about the Earth

Amend the code to correct the **two** errors. [2 marks]

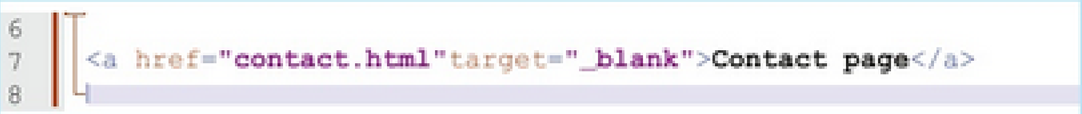
3. A web page needs to have an iframe.

The source of the iframe will be https://www.google.com

It must have a width of 275 pixels and a height of 150 pixels.

Write the HTML code to display this iframe. [3 marks]

4. Figure 7.16 shows some HTML code:



```
6
7  <a href="contact.html"target="_blank">Contact page</a>
8
```

▲ Figure 7.16 HTML code

Explain the purpose of the HTML code on line 7. [2 marks]

5. Explain the purpose of the tabindex attribute. [2 marks]

6. HTML is used to structure web pages.

Page elements can be inline or block level.

Explain the difference between inline and block level. [2 marks]

TOPIC 8

UNDERSTANDING THE FUNCTIONALITY OF CSS

CSS (Cascading Style Sheets) is a language used to specify the formatting and positioning of content within an HTML document. CSS, along with HTML and JavaScript, are the main technologies used on the world wide web. CSS separates the formatting and layout from the content of a web page and in doing so provides greater flexibility and consistency. In this topic you will learn how to link the CSS rules to the HTML and how to use CSS to style page content including text, tables and links. You will also learn how to use CSS to create sophisticated layouts that position content on a page and add features such as animations. In addition, you will find out how to create pages that are suitable for different screen sizes.

TOPIC 8 UNDERSTANDING THE FUNCTIONALITY OF CSS

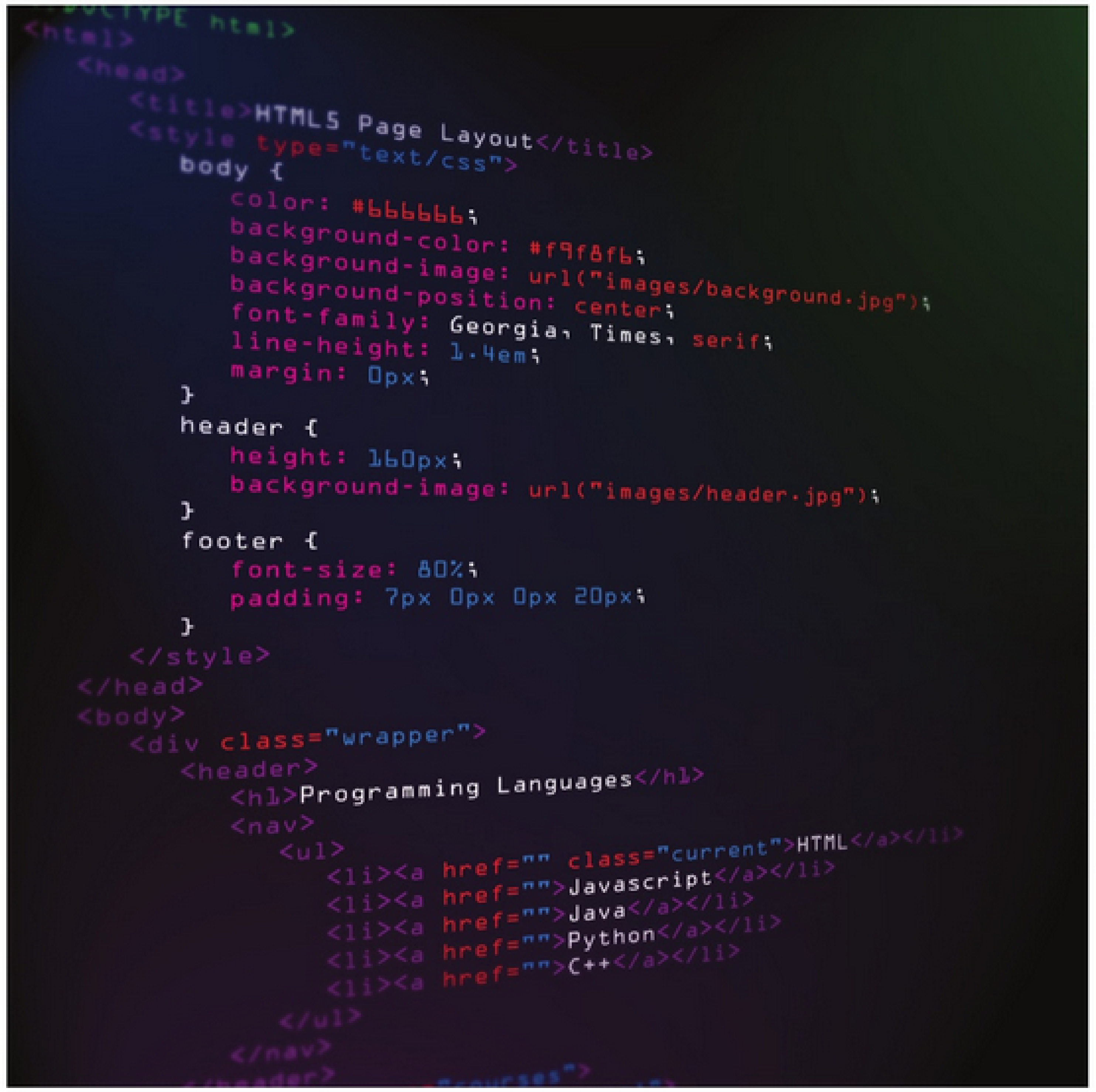
LEARNING OBJECTIVES

By the end of this topic you should be able to understand:

- how CSS can be used within an HTML document
- how to apply styles to a range of HTML elements
- how to position content on a web page
- how to create responsive web pages that can be used with different size screens.

GETTING STARTED

What do you already know about CSS? What is it used for? Is it a programming language or something different? Why are CSS and HTML used together? Do you know any benefits or drawbacks of this approach? Discuss these questions with a partner and see what answers you can find.



```
DOCTYPE html>
<html>
    <head>
        <title>HTML5 Page Layout</title>
        <style type="text/css">
            body {
                color: #666666;
                background-color: #f9f8f6;
                background-image: url("images/background.jpg");
                background-position: center;
                font-family: Georgia, Times, serif;
                line-height: 1.4em;
                margin: 0px;
            }
            header {
                height: 160px;
                background-image: url("images/header.jpg");
            }
            footer {
                font-size: 80%;
                padding: 7px 0px 0px 20px;
            }
        </style>
    </head>
    <body>
        <div class="wrapper">
            <header>
                <h1>Programming Languages</h1>
                <nav>
                    <ul>
                        <li><a href="" class="current">HTML</a></li>
                        <li><a href="">Javascript</a></li>
                        <li><a href="">Java</a></li>
                        <li><a href="">Python</a></li>
                        <li><a href="">C++</a></li>
                        <li><a href=
                    </ul>
                </nav>
```

▶ A screenshot of HTML containing CSS formatting

WRITING CSS

THE PURPOSE OF CSS

CSS is used to style HTML; it is always used in conjunction with HTML and provides control over the formatting and layout of a web page. Formatting includes aspects such as colour, text font, size and spacing of the elements on a web page. Layout covers the positioning of the elements on the page.

One of the benefits of having the HTML and CSS separated is that it helps ensure that the page design (formatting and layout) is consistent across all the pages of a website. In most cases the CSS for a website is defined in one file, which is used across all the pages of the site. This makes sure the formatting and layout is consistent across all the pages, and if a change is needed it can be made in one place rather than in the individual HTML page files (which in a large site could easily number in the hundreds).

KEY TERMS

external style sheet: CSS styles defined in a separate document to the HTML

inline style: CSS styling added to the HTML element tag using the Style attribute

internal style sheet: CSS styles added to the individual HTML document in the head section

property: assigns a particular style to an element

selector: used to select which element the style will apply to

REFERENCING CSS IN HTML CODE

There are several ways that CSS can be linked with the HTML code.

Inline style. You can add CSS styling to a single HTML element. This should really only be used if the element has a unique style not used elsewhere on the page or site. This is because inline styles lose the main benefits of CSS (ease of updating and maintaining consistency). Inline styles are added to the element tag using the style attribute.

Internal style sheet. CSS styles can be defined in the <head> section of an HTML document using the <style> tag. Here the styles for various elements (such as h1–6 headings, paragraphs, tables, etc.) can be defined, which ensures consistency of formatting across the page. However, it does not help with consistency across the whole site. In this topic, internal style sheets are used in the examples, but this is just for convenience.

External style sheet. With this method the CSS is stored in a separate file and linked to each relevant page using the <link> tag in the page headers. This method allows a single CSS style sheet to be used by all the pages in a website and means that if a formatting update is needed, then (as long as no changes are needed to the HTML) it just needs to be done once.

```
<!DOCTYPE HTML>
<html lang="en">
<head>
    <title>My page with external CSS</title>
    <link rel="stylesheet" href="mystyles.css">
</head>
```

This <link> tag in the page header links to the CSS file. The name must match the name the CSS file is saved with.

Folder structure. For a large website with many pages, having a logical folder structure for the web pages, media and CSS is useful to keep files well-organised and easy to find. For example, the CSS file for the pages is probably best kept in a separate folder from the HTML files. Images and other media are also best kept in a separate folder.

Naming conventions. CSS **selector** class and ID names are up to the programmer and can contain upper and lowercase characters and numbers. Using meaningful names with hyphens (so div-style rather than divstyle) helps readability and is consistent with CSS **property** names.

Relative path. The link to the CSS file in the header uses a relative path rather than a URL (starting with http://), rather like an image file path. So if the CSS file is kept in a folder called 'styles' and it is called mystyles.css then the link tag would look like this:

```
<link rel="stylesheet" href="styles/mystyles.css">
```

KEY TERM

value: properties are set to a particular value

WRITING CSS RULES

CSS has a syntax that needs to be followed. Each style has a selector. This identifies the HTML element that the style will apply to. This is followed by a declaration in curly brackets that includes a CSS property and a **value** for that property, separated by a semi colon. For example:

Selector

Property

Value

```
p {
    color:green;
    font-size:11px:
  }
```

Declarations

*Note that CSS and HTML use American spellings:

color instead of colour

center instead of centre.

Selectors

There are many different selectors that can be used and they fall into several categories:

Type – type selectors are the easiest to understand as they simply use the type of HTML element as the selector, such as <td> or <h1>. The example above uses a type selector, with the p (paragraph element) shown.

Class – HTML elements can have a class attribute set to a value and this can be used as a selector for styles. For example, in the HTML document you might use the class attribute with a <div> element, like this:

```
<div class=div-style>
    <p>Text in this division of the page</p>
</div>
```

The styles for the 'div-style' class can be defined like this, with the class name preceded by a period. All text in this class will be green 11px.

```
.div-style {
    color: green;
    font-size = 11px;
  }
```

ID – using the id attribute of the HTML element works in a very similar way to the class selector described above. However, an id is specific to a single element. If you want to use the same style with more than one element, then use a class. Also, an element can only have one id but it can have multiple classes, so you can apply several styles to the same element.

Universal – this selector can be used to select all HTML elements. So for example, on a page where you want all the text to be blue you don't need to add that property to all the styles, you can just use the universal selector, which is identified by the asterisk (*). For example:

```
*{
    color: blue;
  }
```

Attribute – you can use attribute to select the elements to apply a style to. This allows a style to be applied to any element that has a certain attribute. Attribute selectors are covered in detail later in the topic.

Style sheet example

We will look now at a simple style sheet that demonstrates some of the concepts described above. Later in the topic we will look in more detail at text and layout properties and how they are used.

```
<!DOCTYPE HTML>
<html lang="en">
   <head>
      <title>Page title</title>
      <style>
      * {
         font-family: sans-serif;
         }

      h1, h2 {
         color: green;
         background-color: lightgreen;
         padding: 5px;
         }

      h2 {
         text-align: center;
         }
      .more-padding {
         padding: 20px;
         }
      </style>
   </head>
   <body>
      <div>
         <h1>Web page heading</h1>
         <p>Welcome to my web page</p>
         <h2>Heading 2</h2>
         <p>This is a standard paragraph</p>
      </div>
      <div class = "more-padding">
         <p>This has more padding</p>
      </div>
   </body>
</html>
```

The selector for this style is universal (*) so it sets a sans-serif font for all the text on the page.

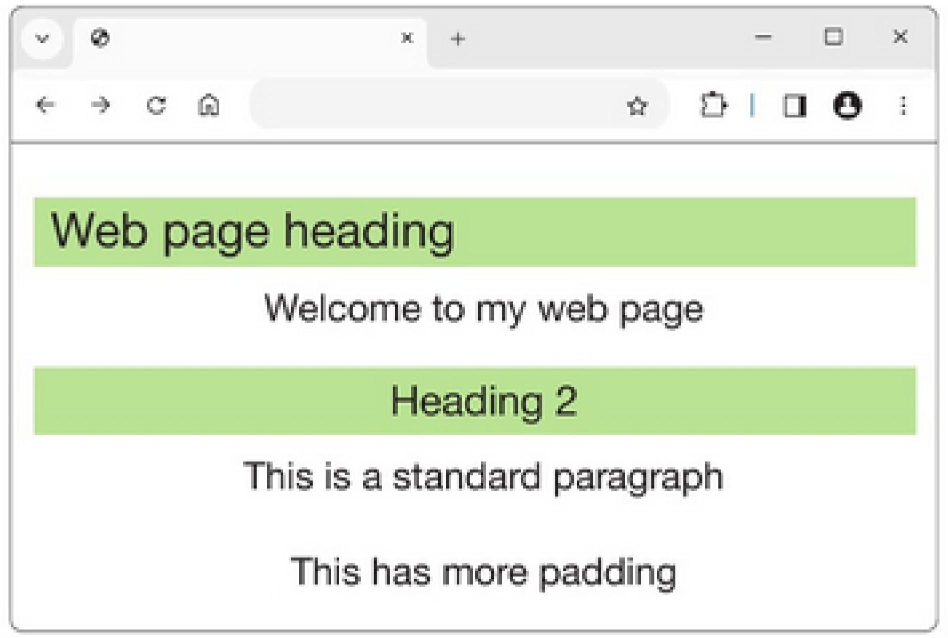
The selectors for this style are the h1 and h2 elements. It sets the text colour to green and the background colour to light green. Padding is space around the element content; it add 5 pixels of padding.

This style adds the text alignment of centre to the h2 style.

This selector allows the style to be applied with a class property.

Here the 'more-padding' style has been applied to this div.

This would look like Figure 8.1 (below).



Web page heading

Welcome to my web page

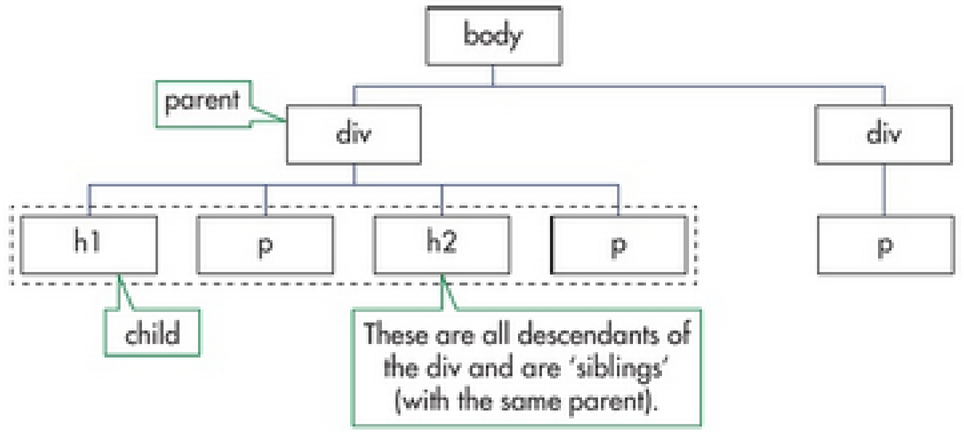
Heading 2

This is a standard paragraph

This has more padding

▶ Figure 8.1 An example of a style sheet being applied.

The way CSS styles are applied is related to the HTML document tree, and the terminology of parents, children, descendants and siblings is used to describe relationships. The document tree for the HTML above is shown in Figure 8.2.


body
parent
div
div
h1
p
h2
p
p
child
These are all descendants of the div and are 'siblings' (with the same parent).

▶ Figure 8.2 The HTML document tree

Combining selectors

You can also create selectors that match related elements using what is sometimes called a **combinator**. This allows quite complex rules for where a style is applied. There are several types of combinators.

KEY TERM

combinator: defines the relationship between selectors when using CSS

Child – a child combinator allows you to select elements that are within (nested with) other elements. For example, suppose you want to apply a style only to h1 elements that are inside a <div> element. The selector would list the 'outer' element <div> followed by a > symbol and the element nested inside it <h1>, like this:

```
   div > h1 {
      color: green;
   }
```

It is important to note that this style will only apply to <h1> elements that are directly related to the <div>, not elements that are nested further down a hierarchy (tree) of elements. The meaning of this will become clearer when we consider what a descendant combinator is.

Descendant – a descendant combinator allows you to select elements that are within (nested with) other elements, but nested anywhere in the hierarchy of styles, not just direct children as with the child combinator. With a descendant, you simply put a space between the outer element and the nested one. Let's look at an example of child and descendant combinators together so we can see the difference.

```
   <!DOCTYPE HTML>
   <html lang="en">
   <head>
      <title>CSS example</title>
      <style>
      div p{
         background:lightblue;
         }
      div>p{
         background:lightgreen;
         }
      </style>
   </head>
```

Descendant combinator, so <p> nested anywhere under a <div>

Child combinator, so <p> directly nested under a <div>

```
   <body>
   <p>Title for page</p>
   <div>
      <p>Text within the division</p>
      <article>
         <p>Text within an article</p>
      </article>
      <p>More text within the division</p>
   </div>
   </body>
   </html>
```

This paragraph is a direct child of the <div> so it will have a green background.

This paragraph is a not a direct child of the <div> because it is nested within an <article> element, but it is a descendant so it will have a blue background.

This paragraph is also a direct child of the <div> so it will have a green background.

Adjacent sibling. This is used to select an element that is directly after another element, not nested within the element, but after it. The two elements are separated by a + sign. Here is an example:

```
   <!DOCTYPE HTML>
   <html lang="en">
      <head>
         <title>CSS example</title>
         <style>
         div p{
            background:lightblue;
            }
         div + p{
            background:lightgreen;
            }
         </style>
      </head>
      <body>
      <p>Title for page</p>
      <div>
         <p>text within the division</p>
         <article>
      </div>
      <p>text after the division</p>
      <p>text in another paragraph</p>
      </body>
      </html>
```

Descendant combinator, as seen above.

Adjacent sibling combinator, so only <p>elements after a <div> will be styled.

This is styled with a blue background by the descendant as before.

This paragraph follows directly after the <div> so it is an adjacent sibling and has a green background.

This paragraph does not directly follow the <div> so it is not styled.

General sibling. This combinator is similar to the adjacent sibling, but the sibling element does not have to directly follow the element. In the HTML above, the last paragraph is not styled by the adjacent sibling, but if the general sibling combinator is used then it would be styled the same as the previous paragraph (green background). General sibling combinators are created using the ~ symbol between the two elements.

So if you changed the CSS div statement for the adjacent sibling in the code above to the general sibling (by changing the + to a ~), the output would change from:

Title for page
text within the division
text after the division
text in another paragraph

to:

Title for page
text within the division
text after the division
text in another paragraph

Multiple selectors. If you want to apply the same styles to several elements, you can simply list the elements as the selector with a comma between them.

SKILLS
ANALYSIS, INTERPRETATION

ACTIVITY 1

1. What are the different roles of HTML and CSS?
2. What are the different ways that CSS styles can be used in a web page?
3. What is the meaning of the terms selector, property and attribute?
4. Combinator selectors are quite difficult to understand. The best way to understand them is to create some example HTML with CSS and try them out. You could start by using the example HTML above for child, descendant and adjacent sibling combinators.

WRITING EFFICIENT STYLE SHEETS

CSS stands for Cascading Style Sheets. Cascading means the way the browser deals with multiple declarations. With the complex CSS that is applied to a sophisticated web page, it may be the case that there are two different declarations that apply different values to the same property. Here are the basic rules of style cascade:

Where there are two declarations for the same selector, the one declared last is the one that applies.

Where the selectors are different but they apply to the same element, then the style that is most specific will apply.

This is best explained with an example.

```
      <!DOCTYPE HTML>
      <html lang="en">
         <head>
            <title>CSS example</title>
            <style>
            .main-title {
               color: red;
               }

            h1 {
               color: blue;
               }
            </style>
         </head>
         <body>
            <h1 class="main-title">This is my heading.</h1>
         </body>
         </html>
```

The selector used here applies to elements with their class attribute set to 'main heading'.

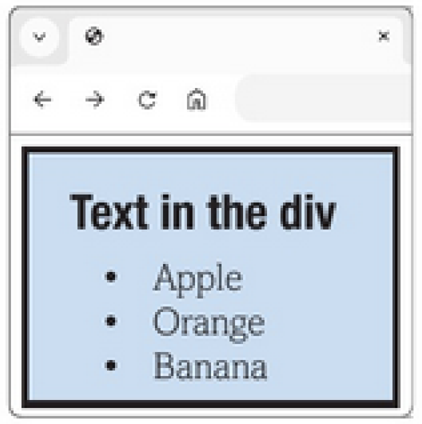
This selector applies a style to the <h1> element.

This creates a style conflict because the <h1> element used here also has its class set to 'main-title'.

In this example there are two styles that can apply to the heading 'This is my heading', one from the style set by the <h1> selector and one from the 'main title' class. The style that is used is the class-based one because it is more specific than the <h1> style. The <h1> style would apply to any <h1> heading element but the class-based style would only apply to those elements that had that class attribute set to 'main-title'. Therefore, the browser selects the more specific style to apply.

Inheritance is another concept that you need to understand. Inheritance defines the way that styles set on an element are passed down to the child elements. Some properties are inherited, and others are not. Where no value is set for an inherited property then the element will inherit that property from its parent.

Properties such as border, margin, padding, etc. are not inherited, whereas all the font properties and colour are inherited. This HTML demonstrates how inheritance works.



Text in the div

- Apple
- Orange
- Banana

▲ Figure 8.3 The code on the right would look like this on screen

```
   <!DOCTYPE HTML>
   <html lang="en">
      <head>
         <title>CSS example</title>
         <style>
         .div-style {
            background-color: lightblue;
            color: blue;
            font-family: sans-serif;
            border: 2px solid black;
            }
      </style>
      </head>
      <body>
         <div class="div-style">
            <h1>Text in the div</h1>
            <ul>
               <li>Apple</li>
               <li>Orange</li>
               <li>Banana</li>
            </ul>
         </div>
      </body>
   </html>
```

This style has colours, font and border set.

The style is applied to this div.

The <ul> element is a child of the div and inherits the colour and font properties. However, it doesn't inherit the border property, otherwise the list would have its own border inside the div border.

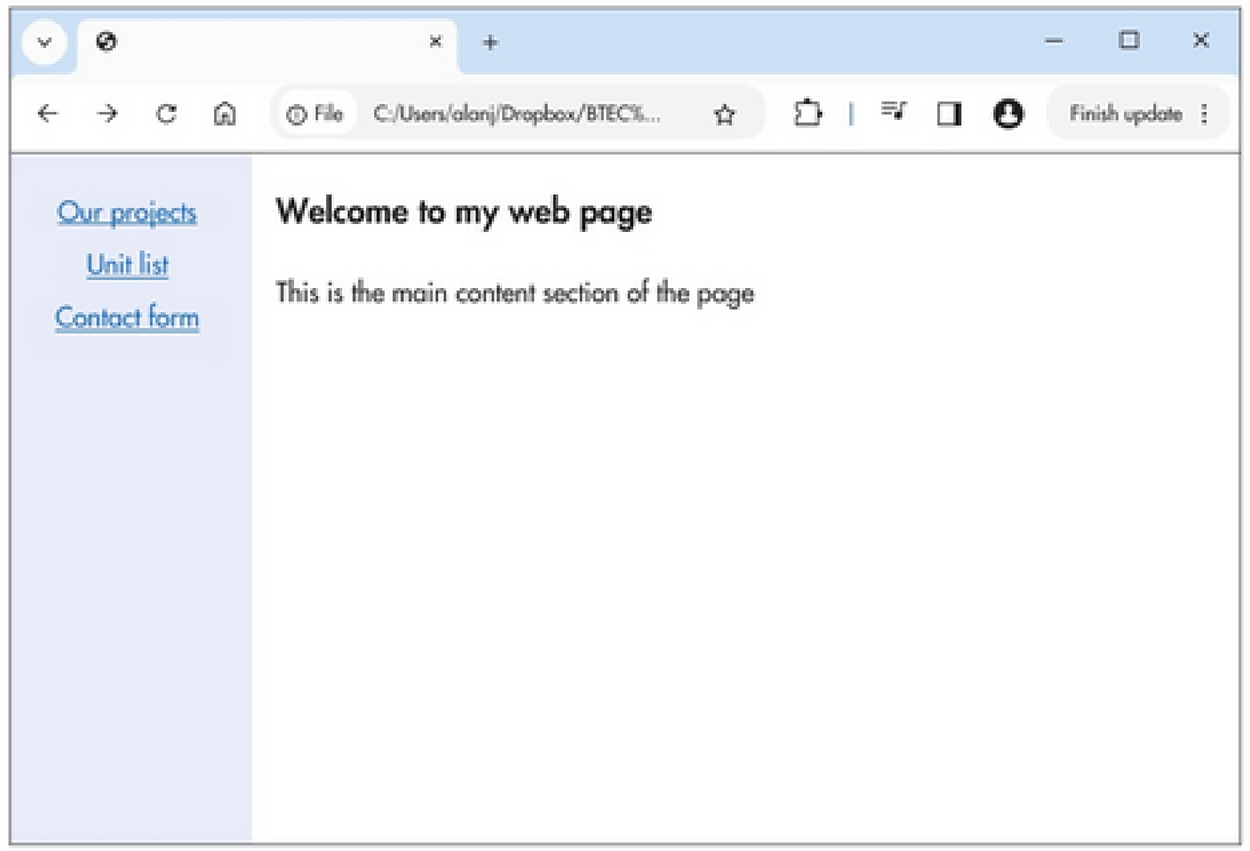
To find out if a CSS property is inherited or not you need to search online for 'CSS property reference'. This will list all the CSS properties (there are about 500 of them!) and state if they are inherited.

UNDERSTANDING THE CSS BOX MODEL

Professional web pages often include a layout in which sections of a page are used for different purposes, such as to provide links and to hold the main content of the page. To do this you need to divide the page up into containers or boxes that have different content. This can be done in CSS using what is known as the 'box model'.

Suppose you wanted to create a layout like the one shown in Figure 8.4, with a sidebar to the left that contains links (or other content), and a main content section on the right.

In order to get the layout properly arranged you need to think about how wide the sidebar will be. This is because you will need to set the left edge of the main content box so it doesn't overlap with the sidebar. Suppose you wanted the sidebar to be 150 pixels wide. You might think that you would therefore set the width to 150 pixels, but the sidebar also needs padding, which creates the gap between its edge and the links. In the sidebar shown, the padding is set to 15 pixels, which creates a 15 pixel gap at the top, bottom, left and right of the sidebar (see Figure 8.5).

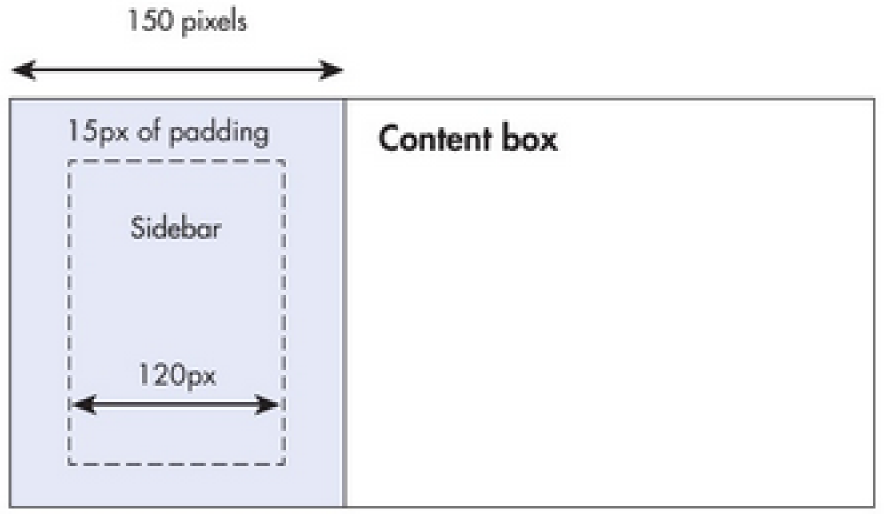


Our projects
Unit list
Contact form

Welcome to my web page

This is the main content section of the page

▶ Figure 8.4 A page layout with a sidebar



▶ Figure 8.5 Calculating box sizes – if the sidebar box is 120px, how does 15px of padding fill 150px?

This padding needs to be added to the width of the sidebar to give a total of 150 pixels. As there is a total of 30px of padding (15 on the right and 15 on the left), the width of the sidebar needs to be set to 120px (to give a total of 150px).

The head section containing the CSS for this layout is as follows:

```
   <!DOCTYPE html>
   <html>
   <head>
   <style>
      .sidebar
         {
         height: 100%;
         width: 120px;
         padding: 15px;
         position: fixed;
         top: 0;
         left: 0;
         background-color: lightblue;
         text-align: center;
         }
```

The height of the sidebar is set to 100% of the page so it runs all the way down the page. The width is set to 120 pixels.

Padding is set to 15 pixels. Top and left positions are set to zero so the bar is in the top left corner of the page.

```
      .content
         {
         margin-left: 150px;
         }
      p, h1
         {
         font-family: sans-serif;
         }
   </style>
   </head>
```

The content box has the name .content and it is given a left margin of 150px. This means it will start where the sidebar ends.

The p and h1 elements are set to a sans-serif font (such as Arial).

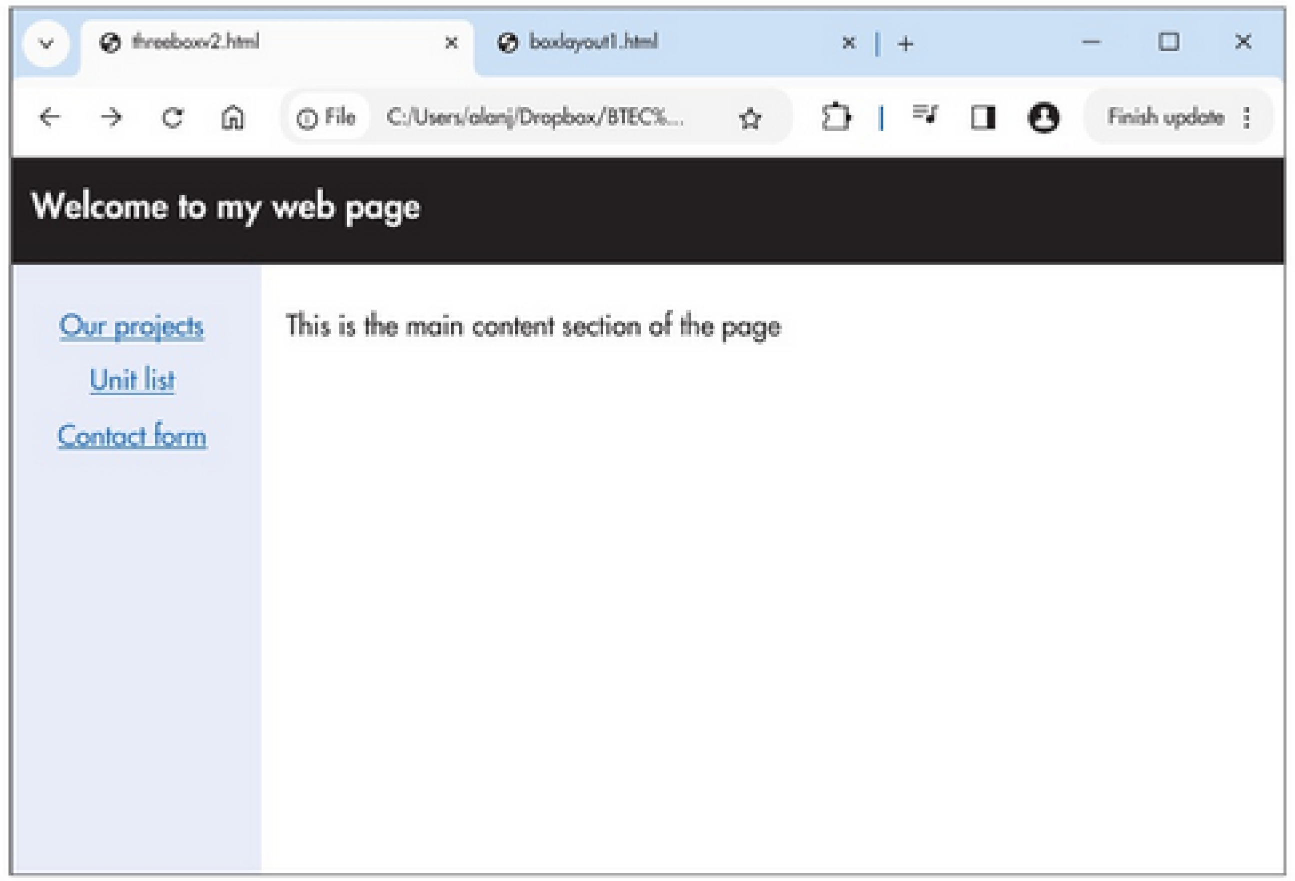
The HTML for the body of this page is as follows (this is in the same file as the head section above, just split here for clarity):

The <div> tag is used to create the boxes on the page using the styles created above. First the sidebar box...

```
   <body>
      <div class="sidebar">
         <p><a href="projects.html">Our projects</a></p>
         <p><a href="table.html">Unit list</a><br></p>
         <p><a href="form.html">Contact form</a></p></div>
      <div class="content">
         <h1>Welcome to my web page</h1>
         <p>This is the main content section of the page</p>
      </div>
   </body>
   </html>
```

Then another <div> tag is used to create the right hand box using the content style.

You are not limited to two boxes for the web page content. You could, for example, extend this layout by adding a title box across the top of the page. To do this you would again need to calculate the sizes for the boxes, taking into account the padding. Figure 8.6 shows an example of how a three-box layout might look, with a title box across the top, a sidebar on the left and a content box.



Welcome to my web page

Our projects
Unit list
Contact form

This is the main content section of the page

▶ Figure 8.6 A three-box layout

Figure 8.7 shows the calculations for the box sizes and positions that are needed to make sure the boxes don't overlap.

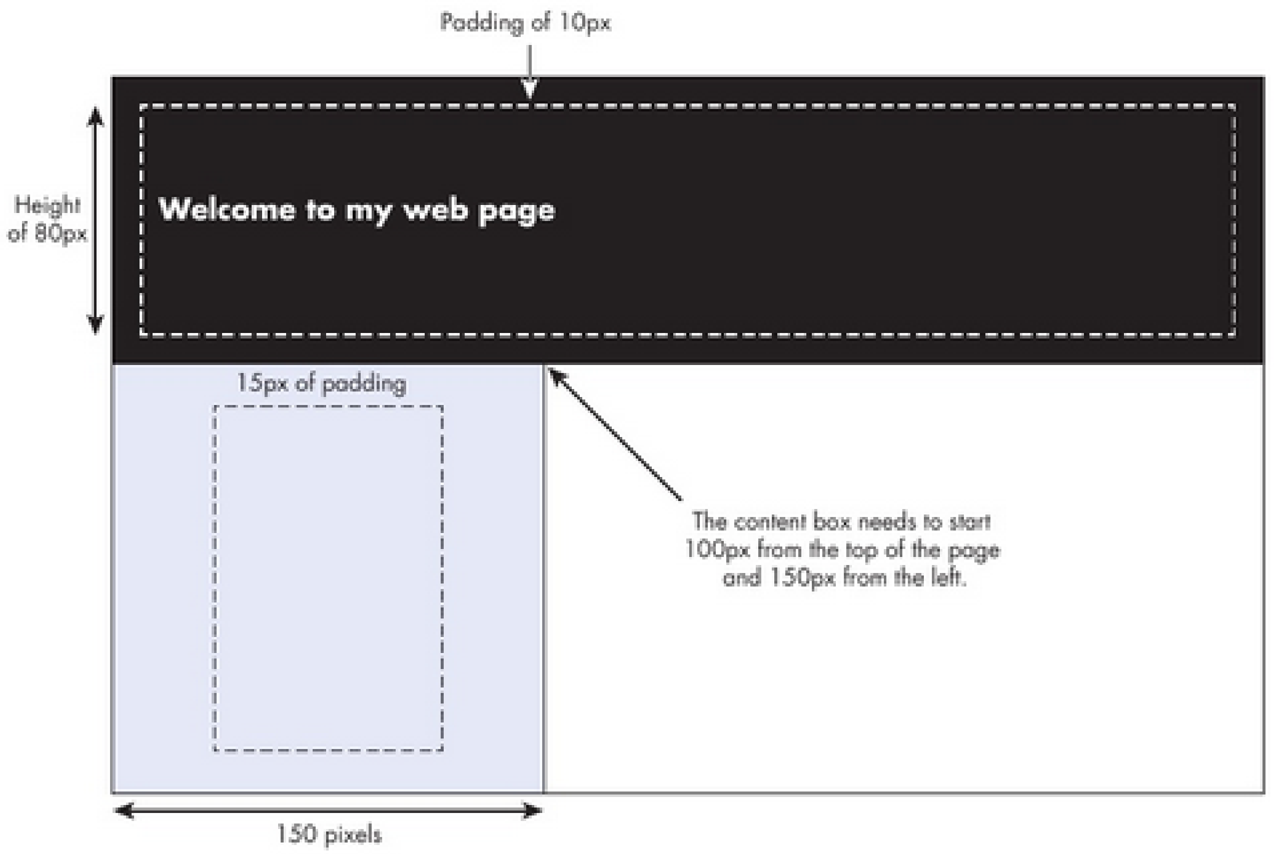

Padding of 10px
Height of 80px
Welcome to my web page
15px of padding
The content box needs to start 100px from the top of the page and 150px from the left.
150 pixels

Figure 8.7 Calculating sizes for a three-box layout

The top title box has a height of 80px and padding of 10px. Therefore, the links sidebar should start 100px from the top of the page (10px + 10px top and bottom padding, plus the 80px height in the title bar). The content box needs to be 100px from the top of the page (so it starts below the title bar) and 150px from the left of the page (so it starts where the sidebar ends).

Here is the CSS for this layout:

```
   <!DOCTYPE html>
   <html>
   <head>
   <style>
   .title    {
      height: 80px;
      width: 100%;
      padding: 10px;
      position: fixed;
      top: 0;
      left: 0;
      background-color: black;
   }
   .sidebar {
      height: 100%;
      width: 120px;
      padding: 15px;
      position: fixed;
      top: 100px;
      left: 0;
      background-color: lightblue;
   }
   .content {
      padding: 10px;
      position: fixed;
      left: 150px;
      top: 100px;
   }
```

This is the CSS for the title box at the top of the page, named 'title'. The box has a height of 80px and a width of 100% (across all of the browser window). It has padding of 10px (giving the box a total height of 100px).

The position property is set to fixed to make sure the box position is fixed relative to the browser window and doesn't change. Top and left are set to zero so the box starts in the top left corner of the window.

The CSS for the sidebar is the same except that the top property is changed to 100px so it starts under the title box.

The CSS for the content box also changes slightly. The top and left properties are changed so it fits under the title box and to the left of the sidebar.

```
   p, h1 {font-family: sans-serif;}
   h1 {color: white;}
   </style>
   </head>
```

The h1 element is set to white colour so it shows up in the black title bar.

The body section of the page is similar to previously except that another <div> tag is added to insert the content for the title box.

```
   <body>
      <div class="title">
         <h1>Welcome to my web page</h1>
      </div>
      <div class="sidebar">
         <p><a href="projects.html">Our projects</a></p>
         <p><a href="table.html">Unit list</a><br></p>
         <p><a href="form.html">Contact form</a></p>  </div>
      <div class="content">
         <p>This is the main content section of the page</p>
      </div>
   </body>
   </html>
```

The <div> tag for the title box.

There are of course many other layout options you could use, and later in the topic we look at an alternative to the box model called grids.

APPLYING STYLES TO ELEMENTS WITH AN ATTRIBUTE VALUE

There are a number of different ways that CSS rules can be applied depending on the attributes that an element has.

Existence. A style can be applied to those elements that simply have a certain attribute (i.e. it exists). For example, you could use the following CSS to style all the <a> elements that have a title attribute. The attribute you want to select is placed in square brackets after the element, like this:

```
   a[title] {
      color: red
   }
```

Equality. It may be that you want a style to apply only if the attribute is equal to a certain value. In which case you include the required attribute value in the selector:

```
   a[title="example"]{
      color: green;
   }
```

Space. The above method will only style the element if there is an exact match to the attribute value, but you can also set the element to be styled if the value is in a list separated by spaces. To do this you precede the equals sign by the ~ character. For example:

```
   a[title~="example"]{
      color: purple;
   }
```

Prefix. You can set a style based on an attribute that is prefixed (preceded) by a value using this syntax:

```
   a[title^="my"]{
      color: orange;
   }
```

Substring. You can have a substring of the attribute value for a selector using the asterisk character (*), like this:

```
   a[title*="xyz"{
      color:purple;
   }
```

Suffix. You can set a style based on an attribute that is suffixed (followed) by a value using this syntax:

```
   a[title$="example"]{
      color: blue;
   }
```

This example shows how these selectors can be used.

```
   <!DOCTYPE HTML>
   <html lang="en">
      <head>
         <title>CSS example</title>
         <style>
         a[title] {
            color: red;
            }
         a[title="example"]{
            color: green;
            }
         a[title^="my"]{
            color: orange;
            }
      </style>
      </head>
      <body>
      <p>
         <a href="newpage.html" title="anything">First link</a>
         <br>
         <a href="newpage.html" title="example">
           Second link</a>
         <br>
         <a href="newpage.html" title="mypage">
           Third link</a>
         <br>
         <a href="newpage.html">Other link</a>
      </p>
   </body>
   </html>
```

This selector will style any <a> element with a title attribute.

This selector will style any <a> element with a title attribute set to "example"

This selector will style any <a> element with a title that starts with "my".

This text will be red.

This will be green.

This text will be orange.

This text won't be styled.

It is worth noting that the example above has conflicting styles. The first style is applied to any <a> element that has a title attribute, which is all of them, except the last one. As mentioned earlier, when styles conflict they are applied in order. So, if you move the first style in the style sheet to the end of it, then all the links will be in red (except the last one). This can sometimes cause problems with complex CSS as you may not be able to understand why some styles don't seem to be working.

The code above shows how the attribute selectors work, but it doesn't really show why you might want to use them in the CSS of a page. They provide very fine control over exactly where styling is applied and they avoid the need to use ID or class attributes to point to a style. Here are some examples of situations where using attribute selectors might be useful:

When creating forms, the input element has many different types that you might want to use on the form (text, radio, checkbox, date, number, etc.) and you might want to style some of these input types in a particular way. This could be done using an attribute selector, like this:

```
      Input[type="text"] {
         background-color: lightblue;
```

On a page with a lot of links you might want to style certain types of links in a particular way based on the href attribute. Suppose you wanted to style those links that point to an HTTPS secure site in a different way to those that point to a HTTP site. To do this you could use the prefix attribute selector. For example:

```
      a[href^="https"]{
         background-color: green;
      }
```

SKILLS

CREATIVITY, INTERPRETATION

ACTIVITY 2

1. In a situation where there are two style declarations for the same element, how does the browser decide which one is applied?
2. Which symbols are used to create suffix, prefix and substring attribute selectors?
3. Create a box model page layout that includes the following:
   a. a header box that spreads across the full width of the window and is about 150px high
   b. three equal-size content boxes under the header, which are each around 300px high
   c. a footer section under the content boxes similar to the header box
   d. suitable padding and content in each of the boxes.

STYLING WEB PAGES

SPECIFYING COLOURS

We have already seen some examples of applying colour to text and using background colours. Now we will look in more detail about how to specify colours in CSS.

Colour names. As we have seen, you can specify colours using colour names that make it clear what colour you are using. However, you need to know which colour names are supported. There are 140 supported names; the easiest way to check which ones you can use is to search online for HTML colour names.

Hexadecimal notation. This is more complex than using colour names, but allows any colour to be selected from a palette of about 16 million colours, not just the 140 colours covered by the standard names.

Hexadecimal (hex) colour codes look like this:

```
   color: #245610
```

KEY TERM

hexadecimal: a numbering system using base 16 (rather than the base 10 system we commonly use). It is used in CSS for defining colours, with two hexadecimal digits used to represent the red, green and blue in a colour

KEY TERMS

opacity: refers to how transparent an element is

RGB: stands for Red, Green and Blue, which are the primary colours, and which can be mixed together to make any colour

The numbers in the colour codes relate to the amount of red (first two numbers), green (second two numbers) and blue (last two numbers). The numbers are in hexadecimal, so rather than 0–9 (decimal) they run from 0–F (0–9 then A, B, C, D, E and F).

Black would therefore be:

```
   color: #FFFFFF
```

Blue is:

```
   color: #0000FF
```

White is:

```
   color: #000000
```

RGB values

This method of specifying colours is similar to hex notation, but three numbers specify the amount of red, green and blue (**RGB**) in the colour. The numbers have a range of 0 to 255. Therefore, the colour red would be:

```
   color: rgb(255,0,0)
```

As well as the colour itself, you can also specify a number of colour effects.

MANIPULATING COLOURS

Opacity. The opacity (transparency) property can be set for an element; it is a value between 0.0 (transparent) and 1.0 (fully opaque). You can set the opacity for an image or for the background colour used in a div. For example:

```
   .div-style {
      background-color: lightblue;
      color: blue;
      opacity: 0.4;
      }
```

Note that opacity is inherited, so any text in the div styled above would have an opacity of 0.4 as well as the background colour, making it quite hard to read.

Gradients

You can create a smooth transition between two or more colours using a gradient. There are three types of gradient you can use:

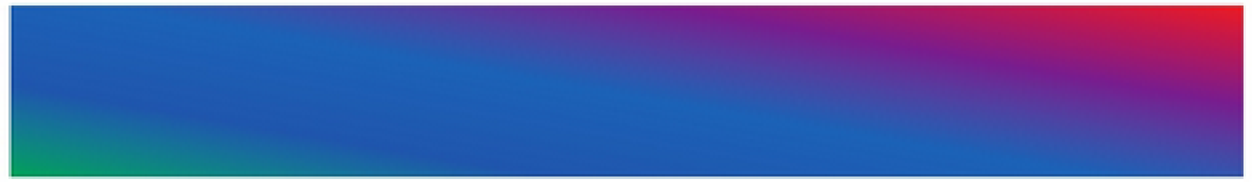
Linear. For this gradient you must define at least two colours, but you can list more colours. A simple example, with a top to bottom (default) gradient, would be:

```
      background-image: linear-gradient(red, green);
```

You can also define the direction of the gradient, and add more colours. For example:

```
      background-image: linear-gradient(to bottom left, red,
      blue, green);
```

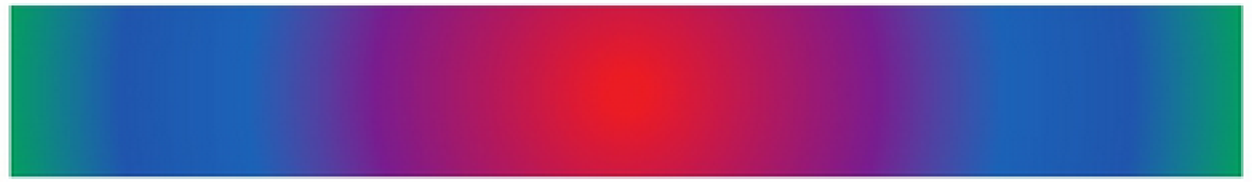
This will produce a gradient from red through blue to green running from top right to bottom left.



▶ A gradient

Radial. With a radial gradient you need to set the start shape (in the centre of the area you are applying the gradient to) to either a circle or an ellipse and at least two colours. For example:

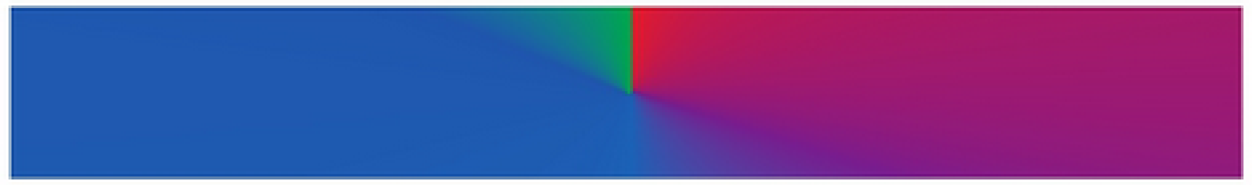
```
      background-image: radial-gradient(circle, red, blue, green);
```



▶ A radial gradient

Conical. A conical gradient rotates the colours around a central point. As with the other gradient you need at least two colours, but you can have more. This example uses three colours:

```
      background-image: conic-gradient(red, blue, green);
```



▶ A conical gradient

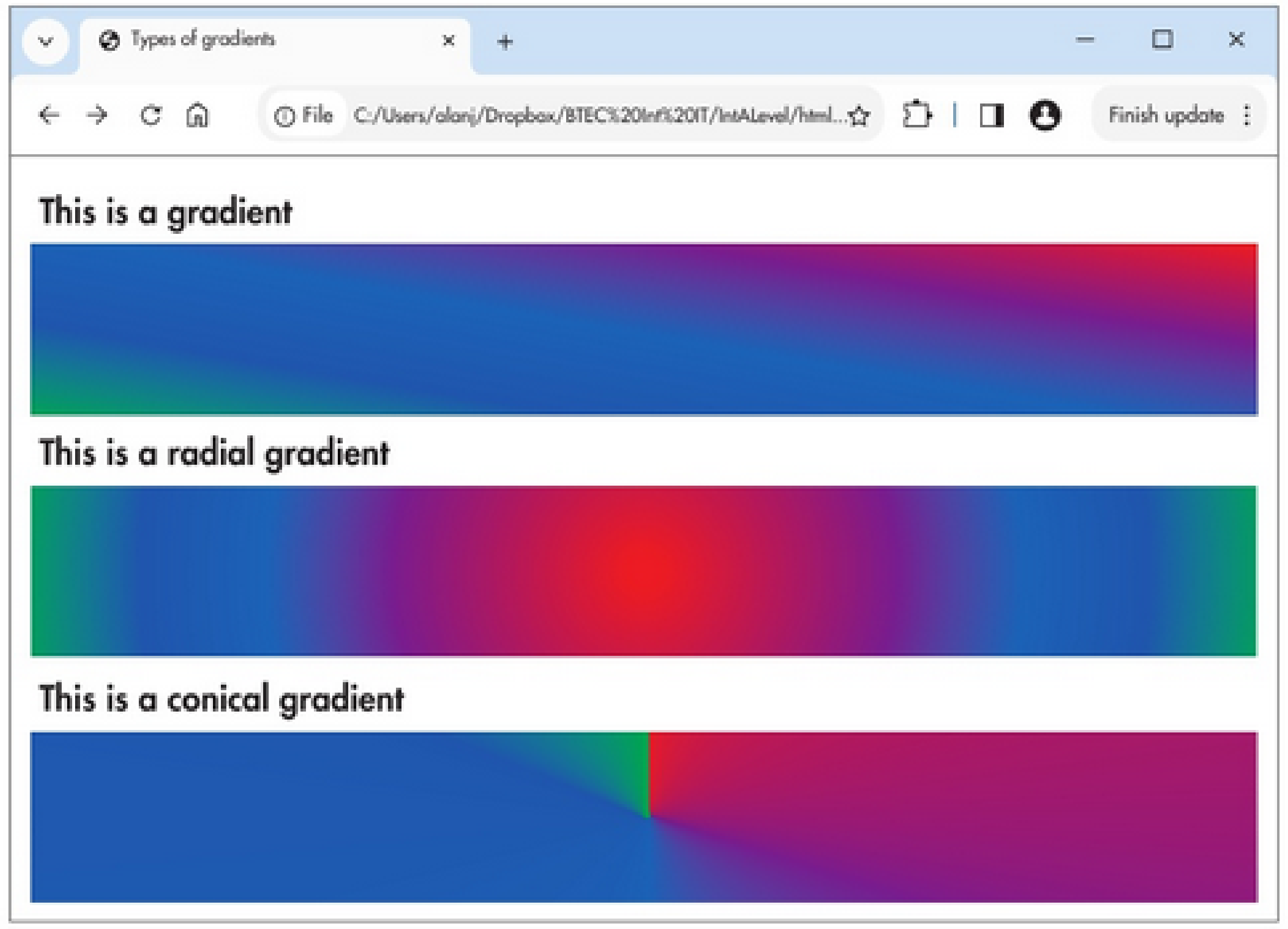
The following HTML code shows examples of the use of all three gradient types.

Styles are created for each gradient type. Because the styles will be applied to a <div> element, a height property needs to be set as there will be nothing else in the div.

```
   <!DOCTYPE HTML>
   <html lang="en">
      <head>
         <title>CSS example</title>
         <style>
         .coloureg{
            background-image: linear-gradient(to bottom left, red,
              blue, green);
            height: 200px;
            }
         .radialg{
            background-image: radial-gradient(circle, red, blue, green);
            height: 200px;
            }
         .conicalg{
            background-image: conic-gradient(red, blue, green);
            height: 200px;
            }
         </style>
   </head>
   <body>
      <h2>This is a gradient</h2>
      <div class="coloureg"></div>
      <h2>This is a radial gradient</h2>
      <div class="radialg"></div>
      <h2>This is a conical gradient</h2>
      <div class="conicalg"></div>
   </body>
   </html>
```

Each style is applied to a <div> using the class attribute.

Figure 8.8 shows this document displayed in a browser.



This is a gradient

This is a radial gradient

This is a conical gradient

▶ Figure 8.8 Examples of different types of gradients

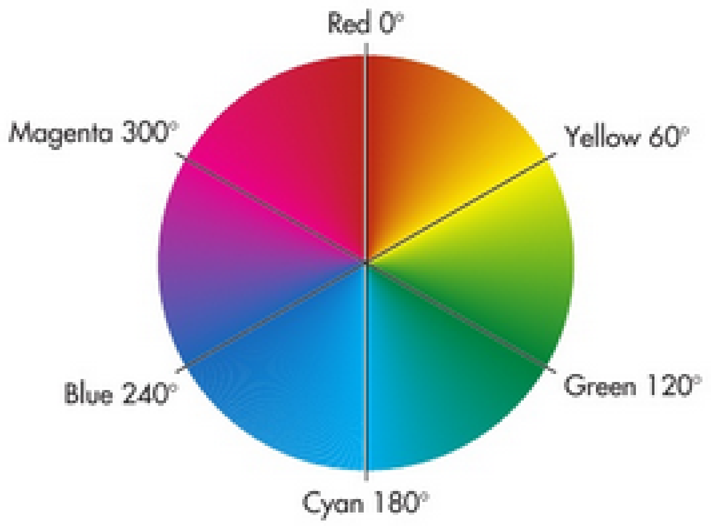
HSL

You can specify the hue, saturation and lightness (**HSL**) of a colour.

KEY TERM

HSL: stands for Hue, Saturation and Lightness and is another way of defining colours in CSS

Hue is the colour as represented on a colour wheel, where 0 degrees is red, 120 degrees is green and 240 degrees is blue.



▶ Figure 8.9 The HSL colour wheel

Saturation is a percentage value that represents the intensity of the colour. A saturation value of 0 would give a light grey colour whatever the hue is set to; 100 would give the full colour intensity.

Lightness is the amount of light you want the colour to have. This is also a percentage: 0 per cent would be black; 100 per cent would be white.

This HTML code shows an example of HSL colours used as backgrounds.

```
      <!DOCTYPE HTML>
      <html lang="en">
         <head>
            <title>CSS example</title>
            <style>
            .hsl-red{
               background-color: hsl(0, 50%, 75%);
               height: 150px;
               }
            .hsl-blue{
               background-color: hsl(240, 100%, 50%);
               height: 150px;
               }
            .hsl-magenta{
               background-color: hsl(300, 25%, 60%);
               height: 150px;
               }
            </style>
         </head>
         <body>
            <div class="hsl-red">
            </div>
            <div class="hsl-blue">
            </div>
            <div class="hsl-magenta">
            </div>
         </body>
      </html>
```

The colour settings give a very wide range of possibilities and can be difficult to visualise from a written description, so the best way to understand them and get familiar with the effects they produce is to experiment with them.

LENGTH VALUES

The CSS length value is used in many properties, and there are several different units that can be used to express length. For example, there is a margin property for top, bottom, left and right, each of which can be set to a different length value. This example shows a range of different measurement units. This is not recommended; it is just an example of how different units are used.

```
   p {
      margin-top: 150px;
      margin-left: 10cm;
      margin-right: 14pt;
   }
```

px is pixels

cm is centimetres

pt is point

Point is a unit used in typography to measure the size of fonts. 1 point is 1/72nd of an inch (~0.352 mm).

These are what are called **absolute** lengths, in that they are a fixed length and do not adjust to screen size. **Relative** lengths, however, change in relation to other values. Relative length includes:

- percentage – relative to the size of the parent element
- vw – relative to viewport width
- vh – relative to viewport height
- em – relative to the current font size (2em is twice the current font size).

Viewport is the user's visible area of a web page.

STYLING ELEMENTS

Styling text

CSS has a lot of properties for styling text. Here we cover just the most common text properties. These can use the body, heading, division and paragraph elements to style all the elements using that element type, or you can create individual styles and apply them to a particular element using a class attribute.

Text colour. We have seen this used earlier in the topic (page 160).

Background colour. This has also been used earlier (page 168).

Alignment. This sets the horizontal alignment of text, with the default being left aligned. Text can also be right aligned, centred or justified (justified text has the letter spacing adjusted so each line ends on the right in the same place). You set the alignment like this:

```
   h1 {
      text-align: left;
      }
   h2 {
      text-align: right;
      }
   h3{
      text-align: center;
      }
   p {
      text-align: justify;
   }
```

Note the US spelling of center.

Line height. This is used to adjust the spacing between lines of text. Most browsers set a default line height of about 1.2 times the text height. A line height of 1.5 will give half a line spacing while a line height of 2 will give double line spacing.

Fonts. Choosing the fonts to be used on a website is important because the style of font can impact on the look and character of the website. You also need to be aware that not all fonts may be available in the browser and operating system being used. With CSS the property used to set text font is 'font-family'. To ensure compatibility several font names should be listed, with the font you really want listed first; end with a generic font family (serif or sans-serif). This allows the browser to choose the font that is available, starting with the first one listed.

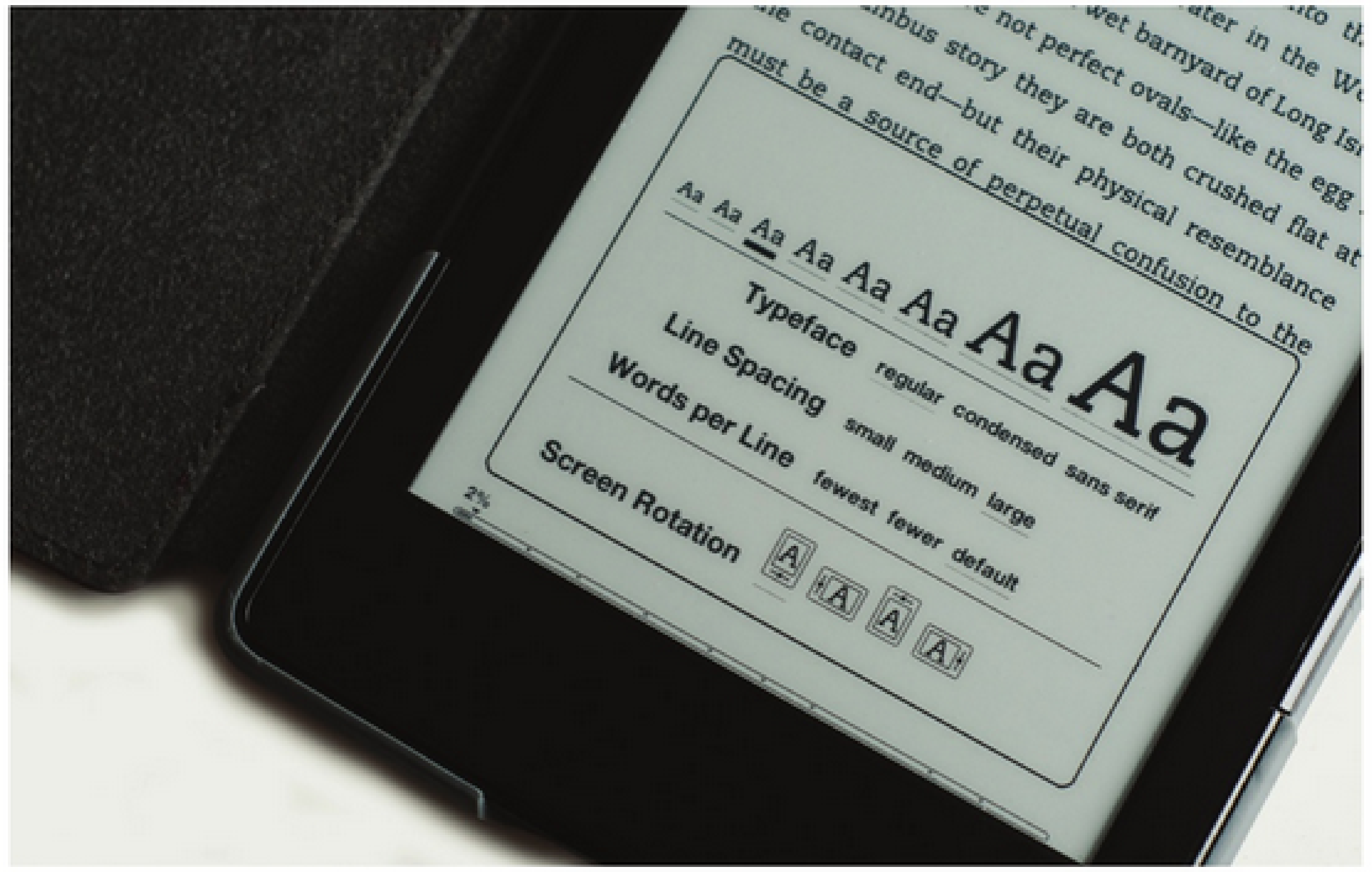
The **font-size property** is used to set the size of the text. As with lengths you have a choice of absolute and relative measurement units. Using **absolute size** such as pixels (px) will mean the text is always that size regardless of the screen size. It also means that users cannot change the text size in some browsers, which is bad for users with visual impairments. Using a **relative size** for fonts, such as em, sets the font size relative to other elements and allows users to change the size if needed; 1em is equivalent to the current font size. The default font size in most browsers is 16px.

KEY TERM

absolute size: a size that does not change and will remain constant

KEY TERM

relative size: a size that is relative to another element such as a parent element, and will change if the size of that element also changes


Aa Aa Aa Aa Aa Aa Aa Aa Aa
Typeface regular condensed sans serif
Line Spacing small medium large
Words per Line fewest fewer default
Screen Rotation

▶ There are different font and other screen options available for use with e-readers

This code shows an example of setting font and size in styles.

```
   .para1 {
      font-family: Arial, Helvetica, sans-serif;
      font-size: 1.5em;
      }
   .para2 {
      font-family: "Times New Roman", Times, serif;
      font-size: 2em;
      }
```

Styling links

Links using the <a> tag can be styled using CSS in several different ways. The font, size and colour of a text link can be styled as already described.

The status of the link (unvisited, visited and so on) can be styled with different colours from the default. There are four link states that you can style in CSS:

- **a:link** – normal link that has not been visited
- **a:visited** – a link that has been visited
- **a:hover** – when the mouse is moved over the link
- **a:active** – when the link is being clicked.

You can, if you want, remove the underline from links using the text-decoration property set to 'none'. This code shows how styles can be created for links:

```
   a {
      text-decoration: none;
      }
   a:link {
      color: green;
      }
   a:visited {
      color: red;
      }
   a:hover {
      color: orange;
      }
   a:active {
      color: blue;
   }
```

Styling lists

With CSS you can set different types of list bullets (for unordered lists) and list numbering (for ordered lists). You can also define an image to be used for a bullet point.

You can use the list-style-type property to change bullets and numbering. For bullets you can use values such as disc, circle and square. For example:

```
   ul.disc {
      list-style-type: disc;
      }
   ul.circle {
      list-style-type: circle;
      }
   ul.square {
      list-style-type: square;
      }
   ol.lroman {
      list-style-type: lower-roman;
      }
   ol.ualpha {
      list-style-type: upper-alpha;
      }
```

Styling tables

Using CSS you can change the basic features of HTML in various ways.

Table borders. With CSS you can set the thickness of the border and also remove the default double border that appears around the whole table and each cell. You can also use the padding property to adjust the space between the table border and the content. For example:

```
   table {
      border-collapse: collapse;
      }
   table, th, td {
      border: 2px solid green;
      padding: 10px;
      }
```

This removes the double border from the table.

This makes the border 2 pixels wide, a green colour and solid.

This adds 10 pixels of padding between the borders and the content.

Width. By default, an HTML table will be sized to fit the content, which can result in the table looking quite small. Using CSS you can control the size of the table with the width property. In this example, the width of the table is set to 50 per cent of the window.

```
   table {
      border-collapse: collapse;
      width: 50%;
      }
```

Text styling. You can use many of the text styling properties that have already been described in tables. These include color, background-color (US spelling) and text-align.

Styling forms

As with tables, forms can have the text styled using many of the text properties already covered. These can be applied to the whole table or, using attribute selectors as described earlier, to individual types of input elements. Text input boxes can be styled with the padding, width and margin properties:

Padding is extra space added inside the input box; pixels (px) are usually used for this.

Width: set how wide the input box is; percentage is usually used, which sizes the box as a percentage of the width of the window.

Margin: creates space around the input box; set in pixels.

For example:

```
   input[type=text] {
      padding: 10px;
      width: 50%;
      margin: 25px;
      }
```

The border property can be used to style borders around a text input box.

Styling images

Images can also have borders added using CSS with the border property. For example, you could add a 4-pixel-wide solid red colour border using the following CSS:

```
   img {
      border: 4px solid red;
      }
```

THE BOX MODEL

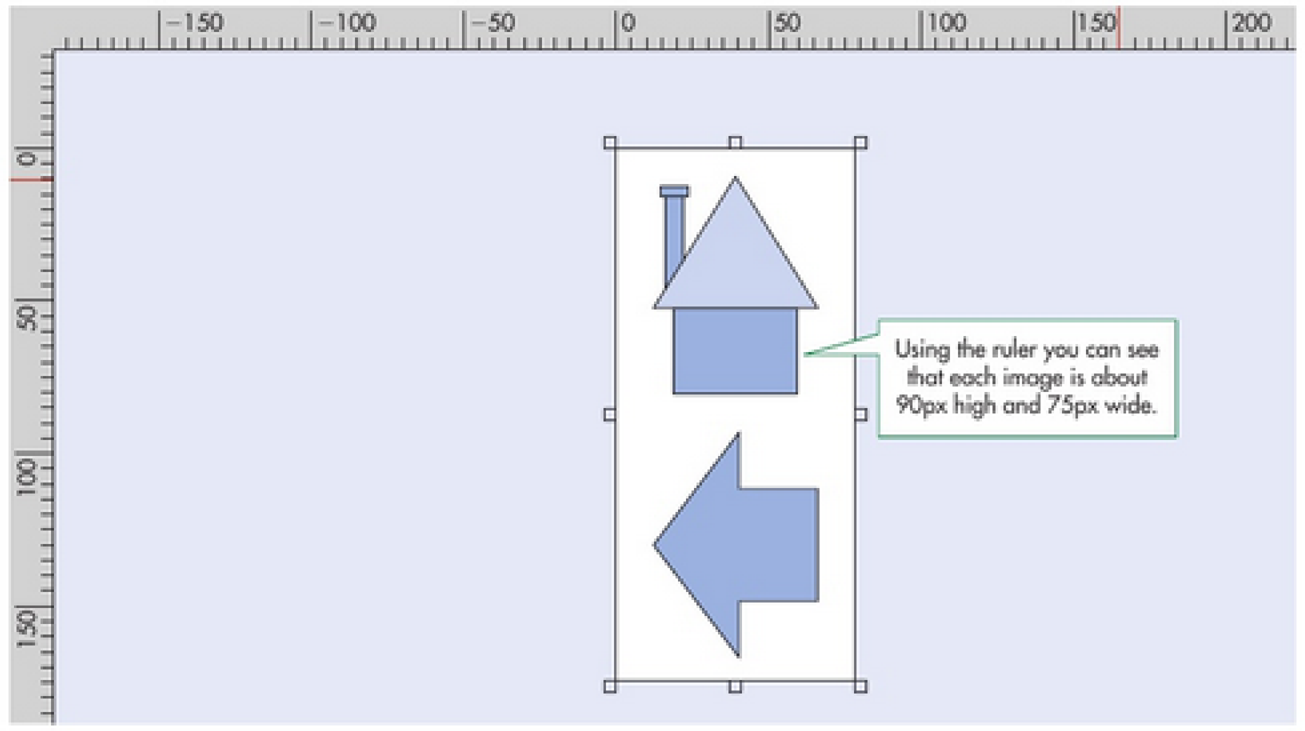
We have already seen an example of the box model, which used the background-color property to set the fill colour of some of the boxes that were created. You can, of course, add borders to boxes and background images if required. We have not added a background image using CSS yet. To do so, you use the background-image property, which takes the image URL as an attribute. The image is placed in the top left corner of the element and is repeated if there is space. Care needs to be taken when choosing background images, as they may obscure other content (such as text) in the foreground of the element. This example shows a box styled with a background image and a border.

```
   .box1 {
      position: fixed;
      top: 100px;
      left: 100px;
      width: 200px;
      height: 200px;
      background-image: url("AL1524936.jpg");
      border: 3px solid blue;
      }
```

KEY TERM

image sprite: an image made up of several images combined in the same file

Image sprites. Sprites are collections of images that are saved in a single file. The use of a sprite reduces the number of download requests a website has to make to the server (just one request for the sprite as opposed to many for each individual image), which can improve the performance of the site. An example of a sprite is shown in Figure 8.10. These images might be used for navigation icons. The image is shown in Microsoft Paint with the ruler turned on, as you need to know the width and height of the individual image within the sprite.



Using the ruler you can see that each image is about 90px high and 75px wide.

Figure 8.10 A sprite image in Paint with the ruler showing

The individual images from the sprite can then be used as shown in this example:

```
<!DOCTYPE html>
<html>
<head>
   <style>
   .home {
      height: 90px;
      width: 75px;
      background-image: url("sprite3.png");
      background-position: 0 0;
      }
   .prev {
      height: 90px;
      width: 75px;
      background-image: url("sprite3.png");
      background-position: 0 -85px;
      }
   </style>
</head>
<body>
   <p>Home icon</p>
   <div class="home">
   </div>
   <p>Previous icon</p>
   <div class="prev">
   </div>
</body>
</html>
```

Two styles are defined which select each icon from the sprite.

The background-position property identifies the horizontal and vertical position. Both images have a horizontal location of 0 but the second image has a vertical position of −85px as it is below the first image.

The sprite images are inserted in the page using div elements.

SKILLS

CREATIVITY, PROBLEM SOLVING, INTERPRETATION

ACTIVITY 3

1. What is hexadecimal colour notation? How is it used? What benefit does it have over other methods of defining colours?
2. What are the four states that a link can be in? How can you style these states?
3. Create an image sprite by combining several images in a single file. Then create an HTML document with CSS and experiment with selecting and displaying the individual images.

POSITIONING CONTENT

POSITION OF ELEMENTS

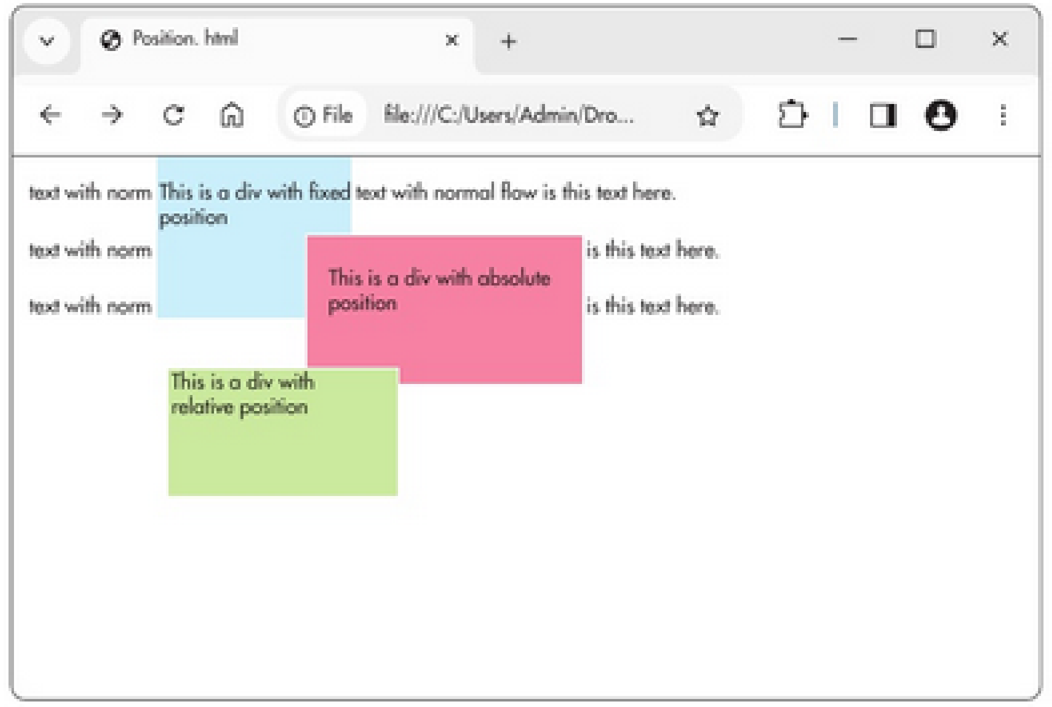
When the CSS box model was introduced earlier, a position property was used and set to fixed. Now we need to look into the position property in a bit more detail. The normal flow of elements on a web page is defined by is the default positioning setting, which is known as static positioning. With normal flow the elements simply follow each other one after another from top to bottom. The top, bottom, left and right properties do not have any effect, so it is not possible to position elements side by side.

Relative positioning. This method of positioning places an element in a position (using the top, bottom, left, right positioning attributes) relative to where it would have been in the normal flow of the page. You could use this, for example, to position an element such as a text input box on a form in a better place than it is positioned using just normal flow.

Absolute positioning. This method allows you to position an element exactly where you want it to be. An element with absolute positioning is not within the normal flow of the page and other elements do not have any effect on an element with absolute positioning.

Fixed positioning. This method positions elements relative to the browser window. They therefore do not move if the window is scrolled. This can mean that some content can be scrolled behind a fixed element so it can no longer be seen.

The difference between these position attributes is hard to understand from a written description, so it might be easier to use an example. Figure 8.11 shows you what you what the page might look like in a browser.



text with norm This is a div with fixed position text with normal flow is this text here.

text with norm is this text here.

text with norm is this text here.

This is a div with absolute position

This is a div with relative position

Figure 8.11 Example of how different position properties look in a browser

The following CSS and HTML shows how this is done.

```
   <!DOCTYPE html>
   <html>
   <head>
      <style>
      .fixdiv {
         position: fixed;
         height: 100px;
         width: 150px;
         top: 0;
         left: 100px;
         background-color: lightblue;
         }
      .absdiv  {
         position: absolute;
         height: 100px;
         width: 200px;
         top: 50px;
         left: 200px;
         background-color: red;
         }
      .reldiv {
         position: relative;
         height: 100px;
         width: 150px;
         top: 20px;
         left: 100px;
         background-color: lightgreen;
         }
      </style>
   </head>
   <body>
      <div>
         <p>text with normal flow is this text here.
         text with normal flow is this text here.</p>
         <p>text with normal flow is this text here.
         text with normal flow is this text here.</p>
         <p>text with normal flow is this text here.
         text with normal flow is this text here.</p>
      </div>
      <div class="fixdiv">
         <p>This is a div with fixed position</p>
      </div>
      <div class="absdiv">
         <p>This is a div with absolute position</p>
      </div>
      <div class="reldiv">
         <p>This is a div with relative position</p>
      </div>
   </body>
   </html>
```

These three styles create boxes with different position attributes; fixed, absolute and relative.

These paragraphs are not styled, so would be in normal document flow.

Three <div> elements are created using each of the styles.

Note firstly that the boxes created with fixed and absolute positioning sit over the normal flow text. You must therefore be careful if you combine these positioning methods.

The box with relative positioning is positioned relative to the normal flow element, and the top (and left) settings are relative to that element, whereas the other two – the top and left settings – are relative to the top left corner of the page.

The difference between the fixed and the absolute boxes is that if the page is scrolled, the fixed box remains in the same place but the rest of the content (including the absolute and relative boxes) scrolls.

Therefore, if you are creating a page which uses a box model layout, you should probably use fixed or absolute positioning for all the elements on the page. If you just need to precisely position a single element (such as a form input box) within a page that uses normal flow, then use relative positioning.

Floating and overlapping elements

Floating elements. The float property sets how an element is positioned inside the element that contains it. An example of this would be how images appear within a page and how text can be made to wrap around the image. By default, text will not wrap around an image and the image will appear on its own with text above and below it. You can change this with the **float** property. The most commonly used attributes with the float property are right and left. So, for example, you could have any images in a page positioned on the left of the page with text wrapping around them with this style:

```
   img {
      float: left;
      }
```

You can also use the float property with other elements. An example would be to use it with the div element to create two side-by-side columns of content.

Overlapping elements. As we saw earlier, elements can be positioned in a way that they overlap each other. The order they are in (which element is on the top) will depend on the order they are, with the last one defined in the HTML being on top. You can adjust the order of positioned elements using the **z-index** property. The following CSS modifies the fixed position box used in the previous example, so it is in front of the box with absolute position, even though it is declared in the HTML after it.

```
   .fixdiv {
      position: fixed;
      height: 100px;
      width: 150px;
      top: 0;
      left: 100px;
      background-color: lightblue;
      z-index: 1;
      }
```

The z-index property is used to bring the element in front of an overlapping element with either no z-index set or with a lower value set.

CONTROLLING THE APPEARANCE OF BOX MODEL LAYOUTS

We have already looked at the box model and how it can be used to create multiple boxes of content on a page. Within a box, the elements by default will behave following the normal flow layout, with each element inside a box flowing on from the previous one. You can adjust this layout using the **display** property, which has a number of possible attributes:

Block. This will change an inline element to be displayed as if it were a block level element (so it starts on a new line rather than inline with the previous element). So, for example you can use the block attribute to make an <a> element behave like a block element.

Inline. This can be used to change the way a block element displays so it is inline.

Grid. Creates a grid-based layout, which is explained later in this topic.

You can also set the display attribute to use advanced layout options such as flexbox. Flexbox is beyond the scope of this topic, but you can research online how it can be used.

CSS AND DIFFERENT SIZE SCREENS

There are a number of ways that web pages can be laid out to accommodate different screen sizes.

Fixed width layout. With this approach, the width of the page is set to a fixed value and this does not change depending on the size of screen that the page is displayed on. Fixed width layouts are created by setting the width of elements in pixels.

Liquid layout. With this approach, the width of the page elements is set as a percentage of the browser window. This means that the elements will scale, depending on the browser window size.

Layout grids

CSS layout grids provide a method of creating a row- and column-based layout without using features positioning as we have done previously with the box model. To create a grid layout, a wrapper or container div element is styled with the display property set to grid. Individual cells within the grid are then styled as required. This code shows an example of a simple three-column grid.

```
   <!DOCTYPE html>
   <html>
   <head>
      <style>
      .grid-wrapper {
         display: grid;
         grid-template-columns: auto auto;
         background-color: green;
         padding: 15px;
         }
      .grid-cell {
         background-color: lightgreen;
         border: 2px solid black;
         padding: 20px;
         font-size: 18px;
         }
      </style>
   </head>
   <body>
      <h1>Grid example</h1>
      <div class="grid-wrapper">
         <div class="grid-cell">Cell 1</div>
         <div class="grid-cell">Cell 2</div>
         <div class="grid-cell">Cell 3</div>
         <div class="grid-cell">Cell 4</div>
      <div>
   </body>
   </html>
```

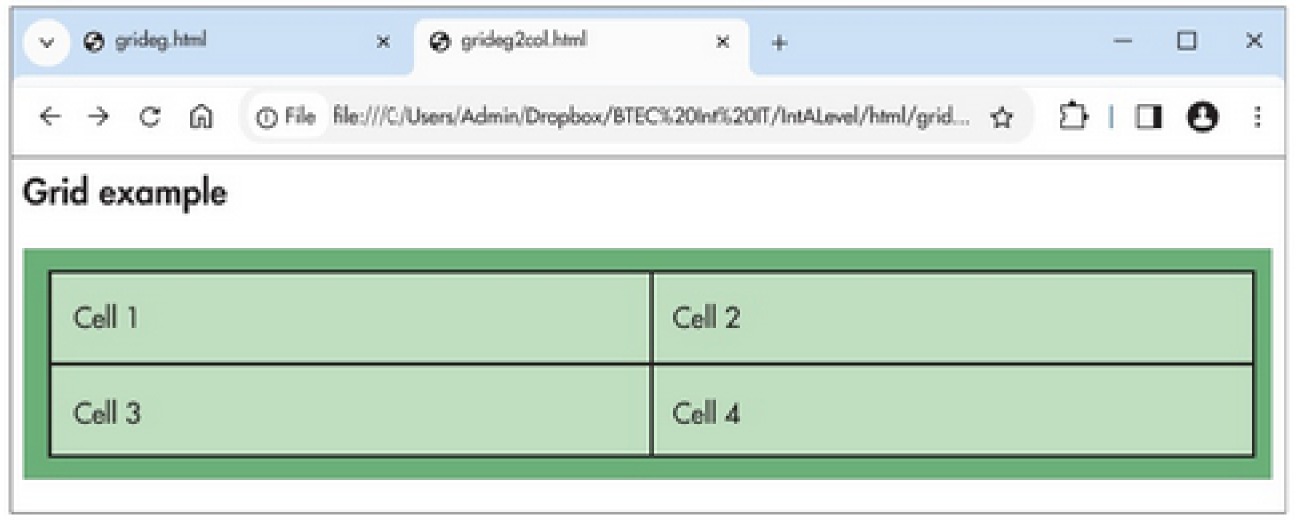
The grid wrapper has the display property set to grid.

This creates a two-column grid with the widths of each column set to auto (fit the screen size).

The grid cell style sets formatting for each cell in the grid.

Here four cells are created, within the wrapper style.

Figure 8.12 shows the grid layout displayed in a browser.


grideg.html
grideg2col.html
File file:///C:/Users/Admin/Dropbox/BTEC%20Inf%20IT/IntALevel/html/grid...

Grid example

Cell 1	Cell 2
Cell 3	Cell 4

▶ Figure 8.12 A simple grid layout

You can, of course, create more complex grid layouts.

This code is an example of a three-column grid, with some styles extending over two rows or columns and different formatting applied to some of the cells.

```
   <!DOCTYPE html>
   <html>
   <head>
      <style>
      .grid-wrapper {
         display: grid;
         grid-template-columns: 20% 40% 40%;
         padding: 15px;
         }
      .cell {
         padding: 20px;
         }
      .header {
         grid-column: 2 / 4;
         grid-row: 1;
         background-color: lightgreen;
         font-size: 28px;
         }
      .sidebar {
         grid-column: 1;
         grid-row: 1 / 3;
         border: 2px solid black;
         }
      .content1
         {
         grid-column: 2;
         grid-row: 2;
         }
      .content2
         {
         grid-column: 3;
         grid-row: 2;
         }
      </style>
   </head>
```

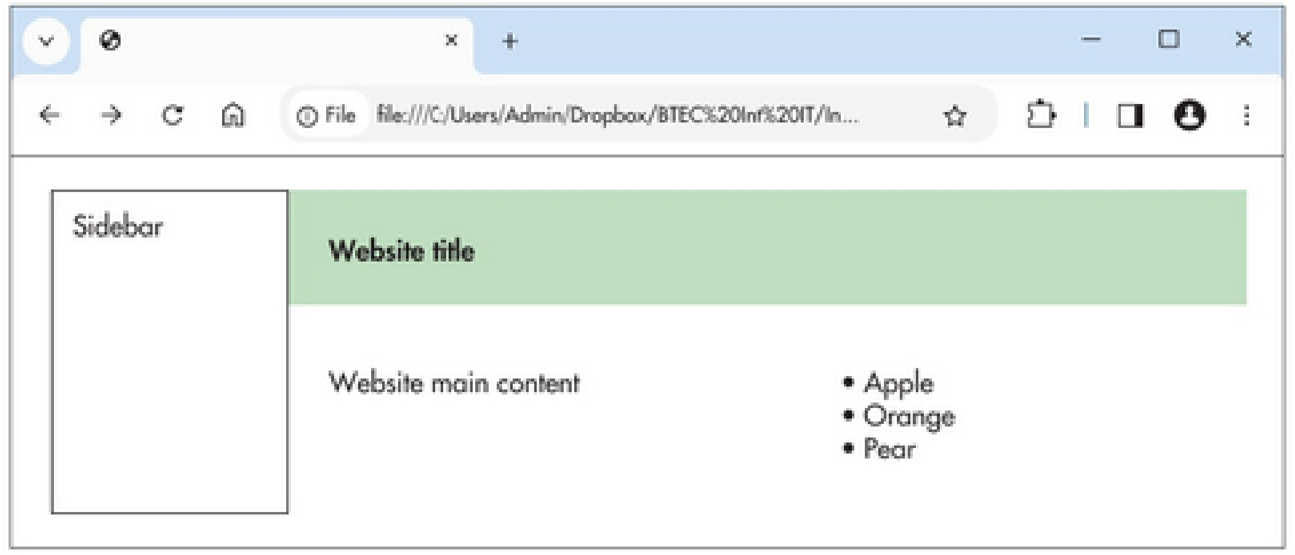
In this example, three columns are defined with different percentage widths.

The grid-column property is used to set the position of the cell; in this case starting in column 2 and ending before column 4.

The grid-row property is used to set the row the cell is in.

```
   <body>
   <div class="grid-wrapper">
      <div class="cell header">Website title</div>
      <div class="cell sidebar">Sidebar</div>
      <div class="cell content1">
         <p>Website main content</p>
      </div>
      <div class="cell content2">
         <ul>
         <li>Apple</li>
         <li>Orange</li>
         <li>Pear</li>
         <ul>
      </div>
   </body>
   </html>
```

Figure 8.13 shows this layout displayed in a browser.



▶ Figure 8.13 A three-column grid layout

CSS frameworks

Frameworks are CSS libraries of styles that are prepared by web developers for others to use. This enables people to create professional-looking pages without having to design and create all the CSS themselves. You can find many examples of frameworks by searching online for CSS frameworks.

KEY TERM

responsive design: website design that responds to different screen sizes and orientations

RESPONSIVE DESIGN TECHNIQUES

Responsive design is about designing and creating a web page that will respond differently when displayed on devices with different screen sizes and orientation. It is very difficult for a site to look good and be easy to use if it uses the same layout on a desktop PC and a mobile phone, so the pages need to adjust to take account of the differences in screens. Responsive design is not a single technology; it is a range of technologies and methods that can be used to help a page adapt. Some examples of responsive design techniques are described below.

Viewport and relative measurements

We have already mentioned that for a page to work well on a variety of screen sizes you would set the viewport using a meta tag in the head section of the page. You should also use relative rather than absolute measurements to create a 'liquid' layout rather than a fixed width one.

Grid layouts

Grid layouts, as described previously, are responsive by their nature and can be further improved by changing the way the columns are defined. The fixed arrangement of rows and columns we have used so far could be made more responsive if the number of rows and columns are adjusted to fit the display window. On a wide screen there would be more columns and fewer rows, but on a narrow screen, such as on a mobile device, the cells should be rearranged into fewer columns and more rows. This is exactly what you can do with the **grid-template-columns** property if you use the **repeat** attribute with the auto-fit and minmax settings. Here is an example:

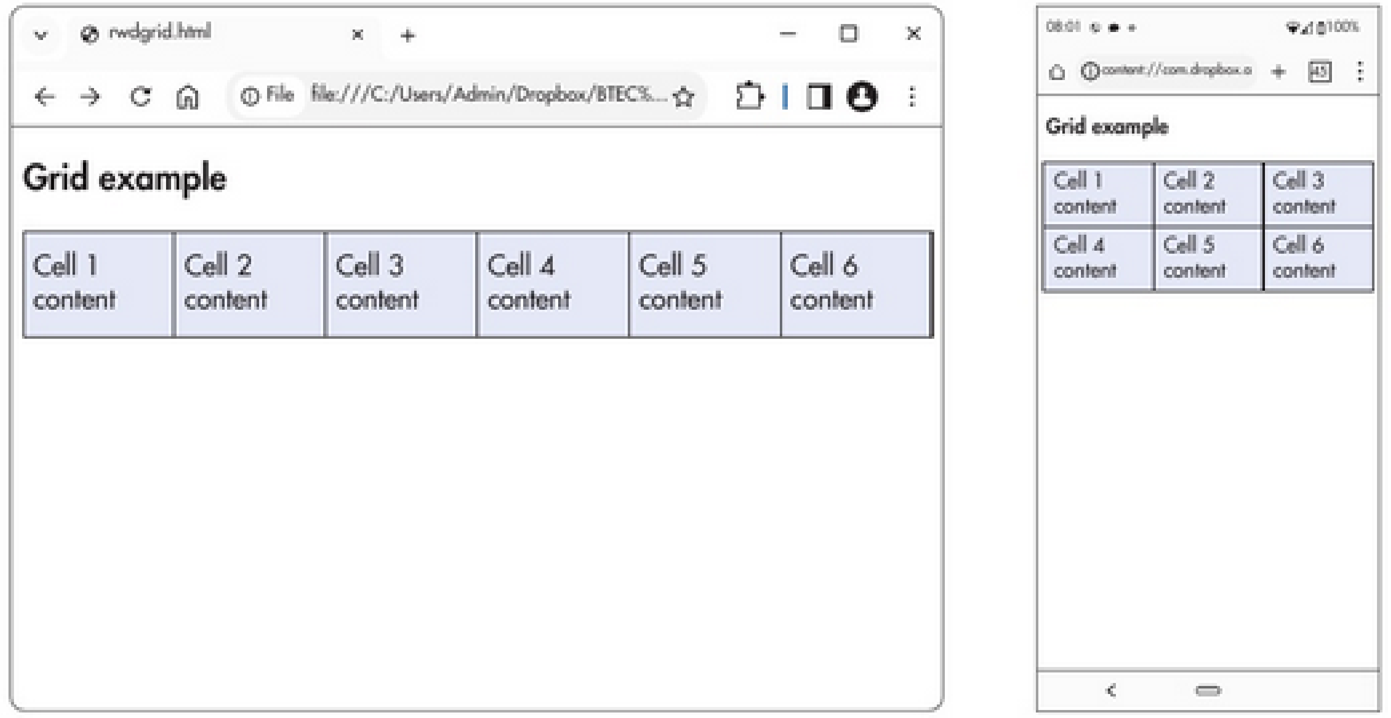
```
   grid-template-columns: repeat(auto-fit, minmax(100px, 1fr));
```

This tells the browser to repeat as many rows and columns as needed and fit them in a minimum width of 100px and a maximum width of 1fr. The fr (fraction) unit is a neat way of telling the browser to automatically divide the number of columns to fit the available width. It provides greater flexibility than using percentages.

This code is an example of a responsive grid layout using this technique.

```
   <!DOCTYPE html>
   <html>
   <head>
   <meta name="viewport" content="width=device-width, initial-scale=1.0">
      <style>
      .grid-wrapper {
         display: grid;
         grid-template-columns: repeat(auto-fit, minmax(100px, 1fr));
         }
      .grid-cell {
         background-color: lightblue;
         border: 1px solid black;
         padding: 10px;
         font-size: 1.5em;
         }
      </style>
   </head>
   <body>
   <h1>Grid example</h1>
   <div class="grid-wrapper">
      <div class="grid-cell">Cell 1 content</div>
      <div class="grid-cell">Cell 2 content </div>
      <div class="grid-cell">Cell 3 content</div>
      <div class="grid-cell">Cell 4 content</div>
      <div class="grid-cell">Cell 5 content</div>
      <div class="grid-cell">Cell 6 content</div>
   </div>
   </body>
```

Figure 8.14 shows this displayed on a desktop PC screen and a mobile device. As you can see, the row and column layout automatically adjusts.



Grid example

Cell 1 content	Cell 2 content	Cell 3 content	Cell 4 content	Cell 5 content	Cell 6 content

Grid example

Cell 1 content	Cell 2 content	Cell 3 content
Cell 4 content	Cell 5 content	Cell 6 content

▶ Figure 8.14 A responsive grid layout

Responsive images

Using fixed sizes for images will not allow them to be scaled to fit the size of the screen. Setting the max-width property to 100 per cent will scale the image to fit the element it is within.

```
img {
    max-width: 100%;
    }
```

Used with a style like the above, an image inserted in a grid cell will always scale to fit the cell size, even if the cell changes size to accommodate different screen sizes.

SKILLS

PROBLEM SOLVING, ADAPTIVE LEARNING

ACTIVITY 4

1. Create a layout with a fixed position and an absolute position box side by side on the page. What are the differences between the two? Adjust the positions of the boxes so they overlap. Which one is on top and why? How can you swap the order of the boxes? Investigate what happens if you add another box with its position set to relative.
2. Create a grid layout with eight cells and arrange them so they automatically reposition if the screen/window size changes. Insert an image in one of the cells and style it so it resizes if the cell size changes.

KEY TERM

transition: a gradual change from one property value to another

CSS ANIMATIONS

TRANSITIONS

There are a number of animated effects that you can apply with CSS.

A **transition** allows a property gradually to change its value. With the transition property you need to provide the CSS property that the transition is to be applied to and the time (in seconds) over which the transition should occur. Here is an example.

```
<!DOCTYPE html>
<html>
<head>
    <style>
    div {
        width: 150px;
        height: 150px;
        background-color: lightblue;
        transition: background-color 3s;
        }
    div:hover {
        background-color: blue;
        }
    </style>
</head>
<body>
    <h1>Transition example</h1>
    <div></div>
</body>
</html>
```

The transition property has the background-color as the property that will change, and 3 seconds as the period over which the transition will occur.

The hover selector will cause this style to be applied when the mouse moves over the div. Because this involves a change in the background colour, the transition will also be applied. The transition will cause the light blue colour to change to blue over a 3-second period.

You can modify the way the transition is done using the **transition-timing-function** property, which can have the following values:

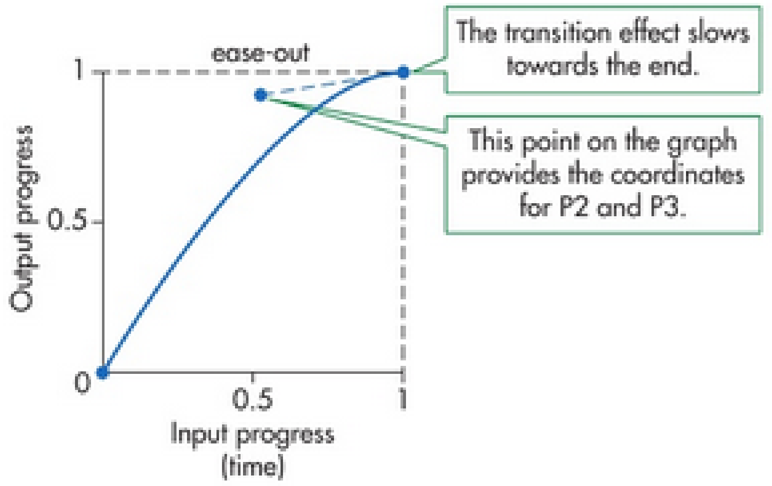
- Ease – this is the default. The effect starts slow, speeds up then slows towards the end.
- Linear – the effect keeps a steady speed.
- Ease-in – the effect has a slow start and then speeds up.
- Ease-out – the effect slows towards the end.
- Ease-in-out – the effect has a slow start and end.
- Cubic-bezier – allows you to specify your own speed values for the effect.

These values can be hard to visualise, so experimenting with them can help you understand how they actually look.

The cubic-bezier value describes a Bézier curve that defines the progress of the transition. Bézier curves are specified by four control points. The first point (sometimes called P0) is the start point and the last point (P3) is the end point. So, for example, look at the following:

```
cubic-bezier(0, 0, 0.5, 0.8)
```

This would produce the same effect as the ease-out setting. Plotted on a graph, the function would look like Figure 8.15.

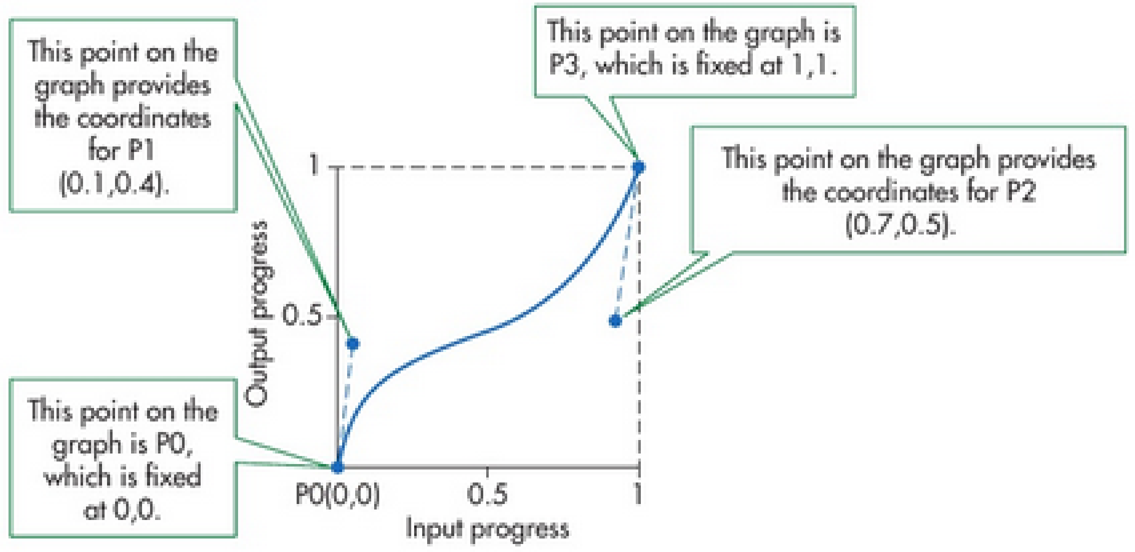


The transition effect slows towards the end.

This point on the graph provides the coordinates for P2 and P3.

▶ Figure 8.15 The ease-out setting, shown on a graph

Suppose you wanted to create a more extreme version of the ease-in-out transition with a very slow start and end. Defined in a graph, it would look like Figure 8.16.



▶ Figure 8.16 Extreme ease-in-out transition

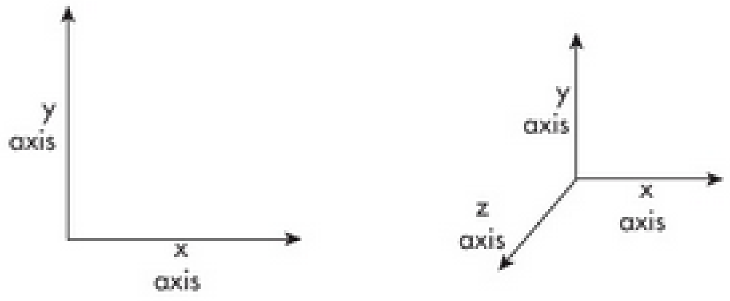
You can find an interactive cubic-bezier graph at cubic-bezier.com where you can experiment with different settings.

KEY TERM

transform: change an element's position and/or orientation

TRANSFORMS

The **transform** property allows you to rotate an element around its x or y-axis (in the case of 2D transformations) or, with 3D transformations, around its x, y or z-axis. Figure 8.17 shows the different axes you can select for the transformations.



▶ Figure 8.17 The axes that an element can be transformed by

2D transforms

These allow you to move, rotate, scale and skew elements. The transform property can have the following methods:

translate(x, y) – moves an element from its current position based on the values of x and y. So:

```
transform: translate(20px, 15px);
```

would move an element 20 pixels to the right and 15 pixels down.

rotate(d) – rotates the element the given number of degrees. So:

```
transform: rotate(15deg);
```

would rotate the element 15 degrees clockwise (use negative values for counter-clockwise).

Scale(w, h) – scales the width and height of an element. So:

```
transform: scale(1.5, 2);
```

will scale the element 1.5 times its width and 2 times its height.

scaleX(w) – scales the width of an element.

scaleY(h) – scales the height of an element.

skew(x, y) – skews the element along the x and y-axis by the specified angles. So:

```
transform: skew(20deg, 30deg);
```

will skew the element by 20 degrees in the x-axis and 30 degrees in the y-axis.

skewX(d) – skews the element along the x-axis by the specified angle. So:

```
transform: skewX(60deg);
```

will skew the element by 20 degrees along the x-axis.

skewY(d) – skews the element along the y-axis by the specified angle.

matrix – allows you to combine a number of transformations. It takes 6 parameters, which are:

- scaleX
- skewY
- skewX
- scaleY
- translateX
- translateY.

3D transforms

These allow you to transform an element in three dimensions. The 3D transformation you can use with the transform property are as follows:

rotateX(d) – rotates an element around its x-axis to a specified angle.

rotateY(d) – rotates an element around its y-axis to a specified angle.

rotateZ(d) – rotates an element around its z-axis to a specified angle.

This code shows an example of a page using both 2D and 3D transformations.

```
<!DOCTYPE html>
<html>
<head>
   <style>
   div {
      width: 100px;
      height: 100px;
      text-align: center;
      left: 50px;
      position: absolute;
      }
   .div-rotate {
      top: 100px;
      background-color: lightblue;
      }
   .div-skew {
      top: 250px;
      background-color: lightgreen;
      }
```

```
         .div-rotate:hover {
            transform: rotateY(25deg);
            }
         .div-skew:hover {
            transform: skew(45deg);
            }
      </style>
   </head>
   <body>
      <h1>Transform examples</h1>
      <div class = "div-rotate">3d rotate me</div>
      <div class = "div-skew">skew me</div>
   </body>
   </html>
```

SKILLS

CREATIVITY, PROBLEM SOLVING, ADAPTIVE LEARNING

ACTIVITY 5

1. What kind of uses could transitions and transforms be put to on a web page? Would having a lot of these animations cause any problems?
2. Create a page that demonstrates all the transition effects covered in this section. The effects can be seen more easily using a transition that takes effect on a change in size of a div. For example, you can start with a div that is 200px wide and then the hover style can change it to 800px wide. Create examples of all the different values you can use with the transition-timing-function property.
3. Create another page that demonstrates all the different 2D and 3D transform properties. Apply them to div boxes using hover styles to apply the transformation.
4. It is possible to create some very simple animations using transform properties, especially if you use an image inside the element you apply the transform to. For example, you could draw an image of an arrow or similar symbol in Paint and then animate it with a transform.

CHECKPOINT

1. What is an inline style?
2. How is the link defined to an external style sheet?
3. Name two types of selector.
4. What is the symbol used for a child combinator?
5. Using the CSS box model, when you require an element with a width of 200px and have padding set to 10px, what width do you need to set for the element to be the correct size?
6. How would you create an attribute selector that selects <a> tags with a title that starts 'abc'?
7. Using hexadecimal colour codes, what would be the code for black?
8. In the HSL colour model, what does the S stand for and what does it control?
9. What does the measurement unit em relate to?
10. What property do you use to style a table without a double border?

THINKING LIKE AN IT SPECIALIST

There are two main aspects to the format and layout of a website. One aspect is the design considerations: how to make the website look professional, sophisticated, and fit with the requirements of the organisation that the website is being developed for. The other aspect is how to create the HTML and CSS code to implement that design. These two aspects require slightly different skills. The design aspect of the site is related more to artistic skills, but the creation of the code is related to IT technical skills. In this topic we have focused on the technical skills. Do you have the required artistic skills to come up with a suitable website design? If you don't have those skills, what can you do about it?

EXAM TIP

There are usually four or five questions in section A of the exam paper on CSS, which follow the first few questions on HTML. The CSS questions will often provide you with some CSS code and ask you to amend or correct it to meet a particular requirement. A question may also ask you to explain how a given example of CSS code works. In section B of the exam paper, you will be given an outline design for a web page in the form of a diagram and asked to implement the design using HTML and CSS.

SKILLS

INTERPRETATION, PROBLEM SOLVING, CRITICAL THINKING

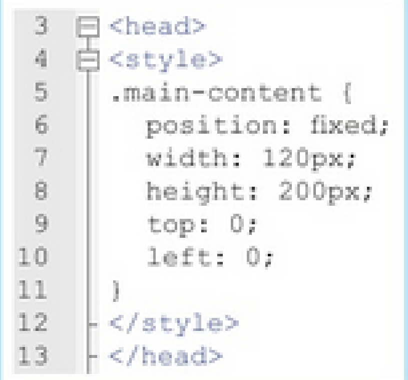
EXAM PRACTICE

1. CSS is used by browsers to format web page content.

   A web page contains text that needs to be formatted. The main heading needs to be 30 pixels in size and blue in colour. Standard paragraph text needs to be aligned justified and have a 10px margin. All text should be in sans serif font. Write CSS declarations of these styles. [3 marks]

2. Describe what is meant when an element is a sibling of another element. [2 marks]

3. Figure 8.18 shows the CSS code used to control the appearance of a shape on a web page.



```
 3  <head>
 4  <style>
 5  .main-content {
 6     position: fixed;
 7     width: 120px;
 8     height: 200px;
 9     top: 0;
10     left: 0;
11  }
12  </style>
13  </head>
```

▶ Figure 8.18 CSS code for a shape on a web page

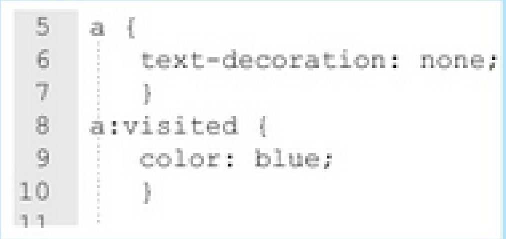
   Describe how the CSS code shown above works with HTML code to position the shape.

   You must include line numbers in your description. [3 marks]

4. CSS code can be referenced within HTML documents by using style sheets.

   Explain **one** difference between an internal style sheet and an inline style. [2 marks]

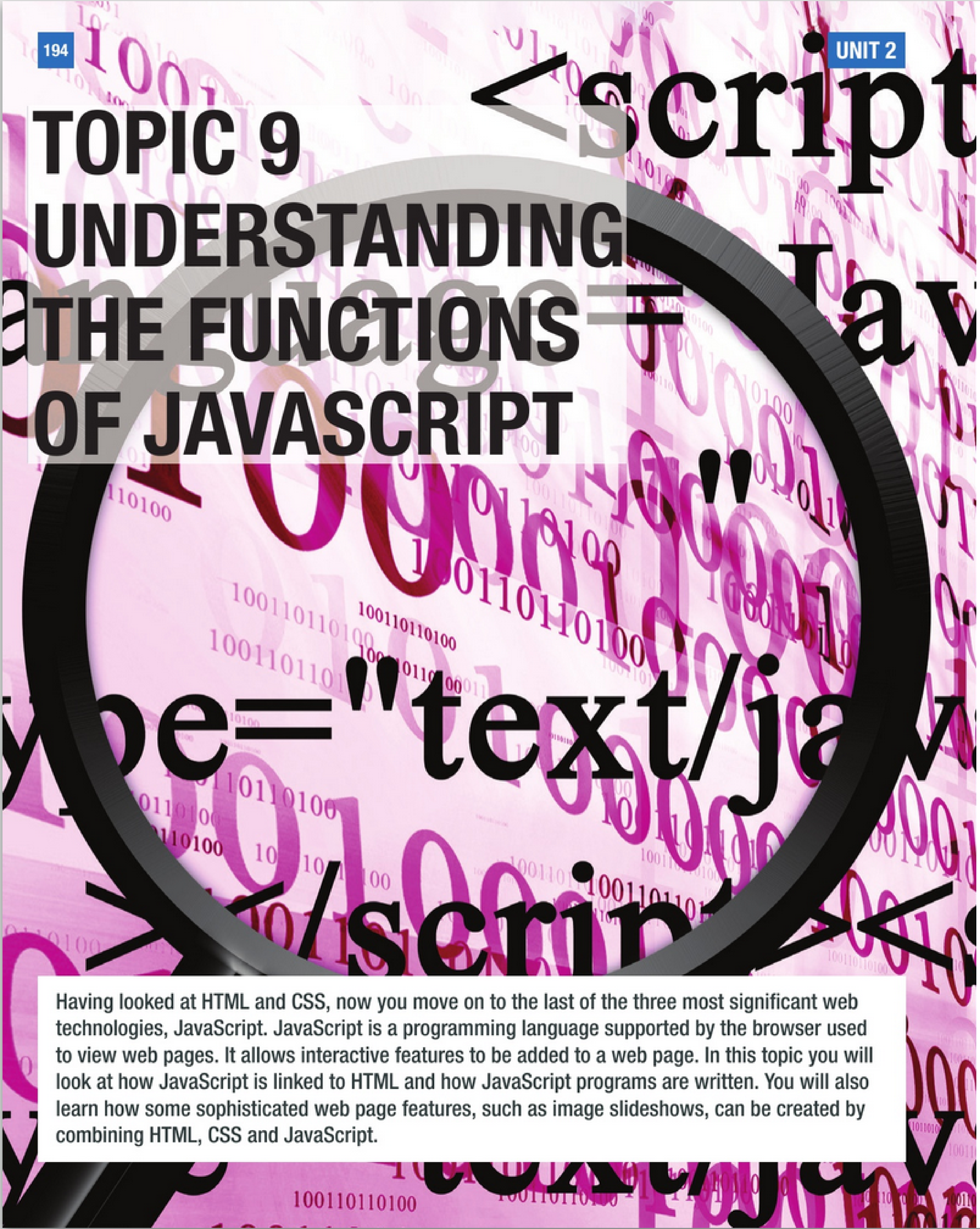
5. Figure 8.19 shows code for a link.



```
 5  a {
 6        text-decoration: none;
 7        }
 8  a:visited {
 9        color: blue;
10        }
11
```

▶ Figure 8.19 Code for a link

   Describe the effect that the CSS code on lines 5 to 9 has on the link. [2 marks]



TOPIC 9 UNDERSTANDING THE FUNCTIONS OF JAVASCRIPT

Having looked at HTML and CSS, now you move on to the last of the three most significant web technologies, JavaScript. JavaScript is a programming language supported by the browser used to view web pages. It allows interactive features to be added to a web page. In this topic you will look at how JavaScript is linked to HTML and how JavaScript programs are written. You will also learn how some sophisticated web page features, such as image slideshows, can be created by combining HTML, CSS and JavaScript.

TOPIC 9 UNDERSTANDING THE FUNCTIONS OF JAVASCRIPT

LEARNING OBJECTIVES

By the end of this topic you should be able to understand:

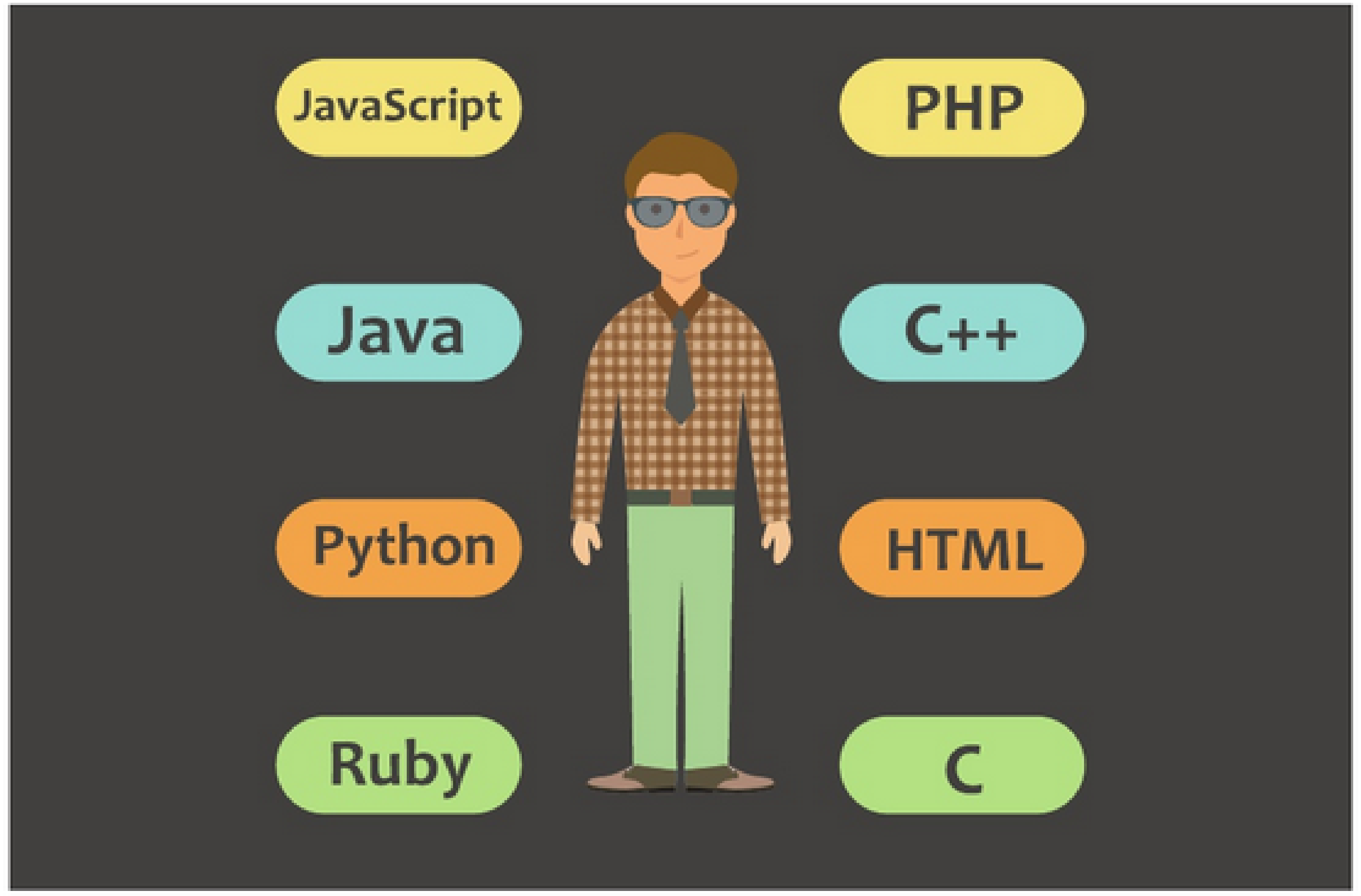
- how JavaScript uses the Document Object Model to interact with HTML
- how to use regular expressions to validate user input
- how to use programming functionality in JavaScript
- how to create web page features using HTML, CSS and JavaScript.

GETTING STARTED

You have probably written some programming code before, but maybe not in JavaScript. Make a list of any programming languages that you have used.

Share your list with a partner. Have you both used the same languages or are there languages that are new to either of you?

▶ A programmer considering a range of programming languages, including JavaScript



KEY TERMS

application programming interface (API): an interface between a programming language and another application that defines how they will communicate

Document Object Model (DOM): an API that gives JavaScript access to HTML

DOCUMENT OBJECT MODEL

JavaScript provides interactivity for web pages so they can respond to user actions in a way that is not possible with HTML and CSS alone. The link between the HTML, which is displayed by the user's browsers, and a JavaScript program is provided by the **Document Object Model (DOM)**. The DOM is an **Application Programming Interface (API)**, which gives JavaScript access to HTML. The DOM represents the HTML document as a tree or map that contains objects that represent the elements in the HTML. JavaScript can therefore access and modify those elements and their attributes (and create new elements if needed), so the structure, style and content of them can be modified. It is also possible to link event handlers to these objects so that when the event is triggered (e.g. a user clicks a button) JavaScript can carry out some action. Although you will be using JavaScript, the DOM is language neutral.

Let's look at some simple examples of how JavaScript can access HTML. There is no need to download any software to create and run JavaScript as it is included in all browsers. JavaScript functions can be placed in either the head or the body section of the HTML. JavaScript in the head section of an HTML document is enclosed in <script> and </script> tags. This is an example of a script that is defined in the head section of the HTML and called from the body.

```
<!DOCTYPE html>
<html>
<head>
   <script>
   function newFunc() {
   document.getElementById("theID").innerHTML = "Modified by
    JavaScript";
   }
   </script>
</head>
<body>
   <h1>JavaScript Example</h1>
   <p id="theID">Paragraph of text</p>
   <button type="button" onclick="newFunc()">Click here</button>
</body>
</html>
```

This function is called newFunc().

It uses the getElementById method to find the HTML element with the ID "theID".

It then changes the text in that element.

This example uses the getElementById method of the document interface and returns an element object whose ID matches the one specified ("theID"). That element is the paragraph of text. The body of the document contains the paragraph of text and a button that is linked to the function using the onclick method. Note that JavaScript is case sensitive, so all the code inside the <script> tags must be typed as shown. For example:

`getElementById` is correct but,

`getElementByID` is incorrect and will not work.

Also, each JavaScript statement ends with a semi-colon (;). Missing off the semi-colon at the end of the statement is a common error. JavaScript errors are not easy to debug as there are no error messages – if there is a syntax or logic error in your JavaScript, nothing usually happens. Therefore, you need to check your code very carefully for things like incorrect case and missing semi-colons.

You can also include JavaScript in the body section of the HTML. Simply use the <script> tags in the body section to enclose the JavaScript code.

In the previous code example, the innerHTML property of the getElementById method is used to write data into the HTML element. This allows JavaScript to be used to modify the HTML on a page, and carry out tasks such as modifying the page layout to suit different screen sizes. There are several other ways that JavaScript programs can output data. For example, the JavaScript alert() method will display an alert box with a message:

```
alert("Input Error!")
```

This is often used for error messages or to provide information to the user.

To obtain input from the user, the most commonly used method is to take this from an HTML form, and a button can be used to trigger a function, as you have seen. Later in the topic you will look at other events that can be used as event triggers. You can also use the prompt() function to display an input dialog box. Prompt can be useful as an easy way to get input into a function for testing purposes.

This code is an example of the use of prompt and alert methods.

```
<!DOCTYPE HTML>
<html lang="en">
<head>
<title>Contact form</title>
<script>
function addup(){
   let x = prompt("Enter a number");
   let y = x*10;
   alert (x + " times 10 is " + y);
   }
</script>
</head>
<body>
   <h1>JavaScript Example</h1>
   <button type="button" onclick="addup()">Click to run function
    </button>
</body>
</html>
```

The prompt simply accepts a number from the user. The input is placed in a variable called 'x'.

The number is multiplied by 10.

An alert is used to display the answer.

In this simple script, a variable is created using the 'let' key word and a value is assigned to that variable using an equals symbol (=). JavaScript variables are covered in more detail later in this topic.

SKILLS

ADAPTIVE LEARNING

ACTIVITY 1

1. Where can JavaScript be located in an HTML document? How do you identify sections of JavaScript code within the document?
2. What's the difference between an alert pop-up box and a prompt?
3. Extend the example shown above so that it allows the user to enter two numbers and then choose to add or subtract them.
4. Test your code. Correct any errors until it works as expected.

KEY TERMS

regular expression (regex): a pattern that describes the format an input should take

validation: involves checking that user input meets the required format or pattern

REGULAR EXPRESSIONS

One common use of JavaScript in web pages is to provide **validation** of data that a user has entered in a form. Sometimes the required data is in a particular pattern of numbers and/or text such as a telephone number or product code. A simple way to validate these patterns is to use a JavaScript feature called a **regular expression (regex)**. To validate the input data you need to create a regular expression that defines what pattern you are looking for. Let's start with a simple example and then build up to something more complex. Suppose you are developing a form where users can enter a product code and then see details of that product. All the product codes follow a defined pattern, which is three uppercase characters followed by four numbers, for example ABC1234. The rules for creating a regex to match a pattern are:

1) The regex must start and end with a forward slash.
2) If a simple pattern is required (e.g. abc) then that can be placed between the slashes, i.e. /abc/. This will match exactly these characters anywhere in the input data.

However, you are looking for any three uppercase characters in the first three characters of the input data. To match any uppercase alphabetical characters, use A-Z in square brackets, i.e. [A-Z]. If lowercase characters are what you are looking for, then use [a-z]. To find three consecutive alphabetic characters, you put the number required in curly brackets, i.e. {3}. This will match exactly three. If you have a comma after the three, it matches three or more.

So far, the regex is: /[A-Z]{3}/

This will match three uppercase alphabetical characters anywhere in the data, but the pattern you are using says they must be at the start of the data. To make sure the three alphabetic characters are at the start of the pattern, you need to add a ^ character at the start:

```
    /^[A-Z]{3}
```

Now you need to add the three numbers to the pattern, which is done in a similar way, adding a $ sign at the end to indicate that the preceding expression must be at the end of the pattern.

```
    /^[A-Z]{3} [0-9]{4}$/
```

Three uppercase alphabetic characters at the start

Four numbers at the end

Now you have the regular expression, you can use it to validate input on a form. This simple example shows how to do this.

```
    <!DOCTYPE html>
    <html>
    <head>
        <script>
        function validateMe() {
            let theInput = document.getElementById("prodID").value;
            let regex = /^[A-Z]{3}[0-9]{4}$/;
            let result = regex.test(theInput);
            if (result){
                window.alert("Good code");
            }
            else{
                window.alert("Bad code");
            }
        }
        </script>
    </head>
    <body>
        <h1>JavaScript validation</h1>
        <form action="" onsubmit="validateMe()">
            <label>Enter product ID </label>
            <input type="text" id="prodID">
            <button type="submit">Submit</button>
        </form>
    </body>
    </html>
```

This function called validateMe will validate the form input.

The function gets the value from the element with the ID of "prodID".

A variable is created for the regex.

The regex method 'test' checks the input against the regex. If it matches it returns true, otherwise, false.

An if function displays an appropriate message.

A form is created runs the validateMe function when it is submitted (onsubmit).

The product ID input box has an ID of "prodID".

There are several other codes that you can use to create a regex, including the following:

Alternatives. To match alternatives use a | character, like this: (a|b|c). You can also use complete words, so /^(north|south|east|west)$/ would allow any one of those four words.

Case insensitive match. Adding a 'i' at the end of the regex will perform case insensitive matching, so /^(north|south|east|west)$/i would match any case combinations of those words.

Digit match. As an alternative to [0-9] you can use \d, so \d{2} finds two digits.

Space match. The code \s matches a space, so /^[A-Z]\s\d/ would match a single uppercase letter followed by a space then a single digit, such as 'Z 4'.

More than one occurrence. To find one or more occurrence of a character or string, use the '+' symbol. So the regex /(a|b)c+/ would match 'ac', 'bc' and 'accc'.

Ending string. To find any string ending in a given string use (?=*string*). So to find any string that ends in .com you would use /[a-z]+(?=\.com). The forward slash is needed to ignore the special meaning of the period character (a period means any character but here you want it to represent a period).

Any characters match. A period represents any character, while an asterisk represents zero or any number of characters. So *[A-Z] would match any number of uppercase characters.

One example of the use of a regex would be to ensure that a user creates a sufficiently strong password. Suppose the rules for the password are:

must contain at least one uppercase character

must contain at leaser one lowercase character

must contain at least one number

the password length must be eight characters or more.

This requires quite a complex regex, so you can break the regex up into a series of rules separated by brackets. The rules are:

^ – start of the regex

(?=.*[A–Z]) – at least 1 uppercase character in the string. Using ?=.* allows the uppercase character to be anywhere in the string

(?=.*[a–z]) – at least 1 lowercase character in the string. ?=.* is used again so it can be anywhere in the string

(?=.*[0–9]) – at least 1 number anywhere in the string

.8, – at least 8 character/digits in length. The comma after the 8 allows 8 or more

$ – end of regex.

Therefore, the complete regex would be:

```
    /^(?=.*[A-Z])(?=.*[a-z])(?=.*[0-9]).{8,}$/
```

SKILLS

ANALYSIS

ACTIVITY 2

1. What regex method is used to find out if a string matches a given pattern?
2. What do the following regex codes match?
   - **a.** /s
   - **b.** ?=
   - **c.** |
3. What regex code would match the following specification (in this order)?
   - **a.** 3 digits (0–9)
   - **b.** A dash (-)
   - **c.** 2 uppercase characters
   - **d.** A final X or Z

Compare the regex code you create with another student. Are there differences? Which one is correct?

KEY TERMS

repetition: a programming construction where a block of code is repeated a certain number of times or until some condition is met

selection: a programming construction where a choice between two blocks of code is made using a condition

variable: a memory area used by a program to store data while the program is running

PROGRAMMED FUNCTIONALITY

So far, you have looked at how JavaScript can be added to a web page, looked at some simple examples of scripts and seen how regular expressions can be used for validation. Next, you are going to look in a little more detail at the functionality of JavaScript.

HOW TO PROGRAM FUNCTIONALITY

Comments

As with any programming code, you are encouraged to add comments to explain what the code is doing. Comments in JavaScript are identified by a double forward slash at the start of the line containing the comment. You can create multi-line comments by starting with /* and ending with */. As well as adding comments, you should use indentation for the blocks of code within your JavaScript. Indentation makes reading code easier and helps identify code blocks used for programming constructs such as **selection** and **repetition**.

Assignment and variables

Like all programming languages, JavaScript uses variables to store data while the program is running. You can automatically create a **variable** by assigning a value, like this:

```
  myVar = 10;
```

However, it is considered better programming practice to declare a variable with the **let** key word, like this:

```
  let myVar = 10;
```

You can also use the **var** key word to create variables, although this is an outdated way of creating variables and is no longer recommended.

If the value you need to store will not change, you should create a constant, using the **const** key word, like this:

```
  const taxRate = 0.23;
```

The equals sign is the assignment operator in JavaScript. Variables and constants have names (identifiers), which must follow certain rules:

- They should start with a letter, but can contain numbers, underscores and dollar symbols.
- They are case sensitive, so myVar and myvar are different variables.
- You cannot use any JavaScript key words as variable names.

It is generally considered good practice to use meaningful names for variables and constants.

KEY TERM

array: a type of data structure, which, unlike a variable that only stores one value, can store multiple values

1D and 2D data structures

You can create a data structure, an **array**, that can store multiple values. This provides a convenient way to store a series of related values rather than having to create individual variables for each one. Each element in the array can be individually accessed using its index number. Index numbers start at zero for the first value in the array. A one-dimensional array is just a list that stores a single row of data items, while a two-dimensional array stores multiple rows.

One-dimensional arrays

These are created using the **const** key word like this:

```
  const myArray = ["apple", "orange", "banana", "mango"];
```

Earlier, the use of the **const** key word to create constant values was mentioned. You might imagine that using **const** to create an array would mean that the values in the array cannot be changed, but this is not the case as it defines a constant reference to the array.

To access an element in the array, you refer to its index number, like this:

```
  let myFruit = myArray[2]
```

This would assign the value 'banana' to the variable myFruit (index numbers start at zero).

There are a number of methods that can be used with arrays. For example, the length property will return the number of elements in an array, like this:

```
  let arrayLen = myArray.length
```

To add or remove items from an array, you can use the push() and pop() methods. So, the following code:

```
  myArray.pop()
```

will remove the last element from the array. To add an item to the end of the array, the push method is used like this:

```
  myArray.push("grape")
```

You can also use the shift() method, which removes the first element in the array, and the unshift() method, which adds an element at the start of the array.

Two-dimensional arrays

These are created by defining an array within an array; this means creating an outer array, which holds an inner array. The syntax for creating a 2D array is similar to a 1D array. Let's look at a simple example. Suppose you wanted to create an array containing items for sale with the number that are in stock. The data might look something like Table 9.1. This is the inner array.

TABLE 9.1: A TWO-DIMENSIONAL ARRAY

Solid state drive (SSD)	Graphics card	Power supply
5	8	3

A two-dimensional array to store this data would be defined like this:

```
const stockArray = [
    ["SSD", 5],
    ["Graphics card", 8],
    ["Power supply", 3]
    ];
```

The array index numbers for this array are shown in Table 9.2. This shows an outer array (shaded) holding the inner array from Table 9.1.

TABLE 9.2: ARRAY INDEX NUMBERS IN A TWO-DIMENSIONAL ARRAY

Column array index

Array index	0	1	2
0	SSD	Graphics card	Power supply
1	5	8	3

Row array index

In this 2D array, an individual element in the array can be accessed by using its column and row index. So, for example:

```
stockArray[2][0]
```

would be "Power supply" (remember, array indexes start at zero).

The following code shows an example of the use of the 2D array, displaying the contents of the array on the page. The example uses a 'for' loop. 'For' loops are described in detail later in this section.

```
<!DOCTYPE html>
<html>
<head>
    <script>
    const stockArray = [
    ["SSD", 5],
    ["Graphics card", 8],
    ["Power supply", 3]
    ];
    function newFunc() {
    let theOutput = "";
    for (let i=0; i<3; i++){
        theOutput += "<br>" + stockArray[i][0] + ", ";
        theOutput += stockArray[i][1] + " in stock";
        }
    document.getElementById("output").innerHTML = theOutput;
    }
    </script>
</head>
<body>
    <h1>2D array example</h1>
    <button type="button" onclick="newFunc()">Click here</button>
    <p id="output"></p>
</body>
</html>
```

2D array

The function uses a 'for' loop to create a string listing the items in the array.

Place the string listing the array contents into a paragraph.

The function uses a loop to change the array index number of the array, but it only does that with the column array index, so it loops through sArray[0][0], sArray[1][0] and sArray[2][0]. Inside the loop it creates an HTML output string to be inserted into the paragraph using its id (which is set to 'output').

TABLE 9.3: LOOPING THROUGH AN ARRAY

'for' loops through columns 0 to 2

Array index	0	1	2
0	SSD	Graphics card	Power supply
1	5	8	3

sArray[0][0]

sArray[1][0]

sArray[2][0]

The row array elements are individually placed in the string rather than using a loop, as explained in the following annotated code.

The variable 'theOutput' is used to build up the HTML statement that will be displayed on the page.

Using the loop counter in the column array index means this will be incremented by the loop.

```
theOutput += "<br>" + sArray[i][0] + ", ";
theOutput += sArray[i][1] + " in stock";
```

The symbol '+=' means 'add this to what is already in the variable'.

The row array index is individually referenced rather than using another loop.

Selection

As you have already seen, the **if** key word is used to implement selection. The structure of an if statement is:

```
if (condition)
{
    Block of code to execute if the condition is true
}
else
{
    Block of code to execute if the condition is false
}
```

The 'else' part is optional and can be omitted if not required. You can also add an 'elseif' part with a new condition that is executed if the previous condition is false.

For example, suppose a calculation of shipping charges needs to be done based on the weight of a parcel. The following code shows an example of how this could be done.

```
<script>
   function parcelCost() {
   let pWeight = prompt("Input parcel weight in grams");
   if (pWeight < 200){
      window.alert("shipping cost is £3");
      }
   else if (pWeight < 500) {
      window.alert("shipping cost is £6");
      }
   else {
      window.alert("shipping cost is £9");
      }
   }
</script>
```

The function collects the weight from the user and assigns it to the pWeight variable.

This statement is executed if the value in pWeight variable is greater than 200.

'else if' is used to add another condition, if the weight is greater than 500.

If neither the 'if' or 'else if' conditions are met then this statement is executed.

This example uses the comparison operator 'less than', which uses the symbol <. The full list of comparison operators is shown in Table 9.4.

TABLE 9.4: COMPARISON OPERATORS

Comparison operator	Meaning
==	Equal to
!=	Not equal to
>	Greater than
<	Less than
>=	Greater than or equal to
<=	Less than or equal to

There are also logical operators that let you combine comparisons. These are shown in Table 9.5.

TABLE 9.5: LOGICAL OPERATORS

Logical operator	Meaning
&&	and
||	or
!	not

You can demonstrate the use of logical operators by modifying the parcel cost function that was shown earlier. Suppose the lowest rate of shipping can only be used if the parcel weighs less than 200 grams AND its length is less than 250 cm. The following code implements this change.

```
<script>
   function parcelCost() {
   let pWeight = prompt("Input parcel weight in grams");
   let pSize = prompt("Input parcel length in cm")
   if (pWeight < 200 && pSize < 250){
      window.alert("shipping cost is £3");
      }
   else if (pWeight < 500) {
      window.alert("shipping cost is £6");
      }
```

The condition now uses the && logical operator to 'and' together the two comparisons.

```
   else {
      window.alert("shipping cost is £9");
      }
   }
</script>
```

Repetition

In many programming situations you need to repeat actions using a loop. In common with most other programming languages, JavaScript provides several ways to implement a loop.

'For' loops

This type of loop repeats a certain number of times. The syntax is:

```
for (initial loop counter; condition for loop to run; counter
 increment)
      {
         Code block to be executed
      }
```

The following code shows an example of the 'for' loop.

```
<!DOCTYPE HTML>
<html lang="en">
<head>
   <script>
      function myFunction(){
         let numb = prompt("Enter number");
         let acc = 1;
         for (let i=0; i <= numb; i++){
            acc += acc*i;
            alert (acc, i);
         }
         document.getElementById("pOut").innerHTML = acc;
      }
   </script>
</head>
<body>
   <h1>Colours</h1>
   <button onclick = "myFunction()">Click here</button>
   <p id="pOut"></p>
</body>
</html>
```

This 'for' loop uses the variable i as a counter; it starts at zero, increments by 1 each time around the loop and continues while the counter is less than or equal to the number the user entered.

In this example, a variable named i is used as a loop counter and is initially set to 0. The loop will continue while i has a value of less than 6, and the value of i is increased by 1 each time around the loop. 'For' loops are best used when you know the number of times to go around the loop.

'While' loops

A 'while' loop continues while a condition is true. The 'while' loop has the following syntax:

```
while(condition)
{
   Code block to be executed
}
```

The following code shows an example of the 'while' loop.

```
   <!DOCTYPE html>
   <html>
   <head>
      <script>
         function divideMe() {
            let theNumber = prompt("Enter a number");
            let theOutput = "";
            while (theNumber > 1){
               theNumber = theNumber / 2;
               theOutput += "<br>value is " + theNumber;
               }
            document.getElementById("output").innerHTML = theOutput;
         }
      </script>
   </head>
   <body>
      <h1>While loop example</h1>
      <button onclick="divideMe()">Go</button>
      <p id="output"></p>
   </body>
   </html>
```

A number is input via a prompt.

A 'while' loop continues until the number is less than 1.

The number is divided by 2.

A list of the results of the division is inserted in the page.

This type of 'while' loop has the condition at the start of the loop, so if the condition is met at the start of the loop, the code within the loop will not be executed at all.

A 'while' loop is best used when you don't know in advance how many times to go around the loop.

A variation on the 'while' loop is the 'do while' loop. With this type of loop, the condition is tested at the end of the loop so the code within the loop is always executed at least once. The syntax of a 'do while' loop is:

```
   do
   {
      Code block to be executed
      } while(condition)
```

Iteration

As well as the repetition methods described above, JavaScript also provides a way you can iterate through an array. The forEach() method calls a function once for each element in the array.

The following code shows an example of the forEach() method.

```
   <!DOCTYPE html>
   <html>
   <head>
   </head>
   <body>
      <h1>For Each loop example</h1>
      <p id="output"></p>
      <script>
         const myArray = ["apple", "orange", "banana", "mango"];
         let theOutput = "";
         myArray.forEach(theFunc)
```

An array is created.

The function is called for each element in the array.

```
         document.getElementById("output").innerHTML = theOutput;
         function theFunc(element){
            theOutput += "<br>value " + element;
            }
      </script>
   </body>
   </html>
```

The function simply builds up a string containing all the array elements.

Subprograms

You have already seen the use of the function key word in JavaScript. It is used to create a mini-program that carries out some tasks, also called a subprogram. You can create any number of functions within the <script> tags and these provide reusable tasks that potentially can be used to add functionality to several different HTML elements. The syntax for creating a function is:

```
   function name(parameter1, parameter2…. ) {
      function code
   }
```

Function parameters are optional, and you haven't used this feature so far. Parameters allow you to pass values to the function, which can help make the function reusable. For example, a validation function could validate several different inputs, or a calculation function could include user input or external values.

Functions that end with a return statement can pass a value back to the code that called the function.

The following code shows an example of a JavaScript function that converts miles to kilometres. It uses two subprograms: function formProcess() and function milesConv(milesIn). This means that the code does not need to duplicate the code within the subprogram whenever it is needed.

```
   <!DOCTYPE html>
   <html>
   <head>
      <script>
      function formProcess(){
         let theInput = document.getElementById("miles").value;
         let km = milesConv(theInput);
         window.alert("kilometres: " + km);
      };
      function milesConv(milesIn) {
         let kmOut = milesIn * 1.60934;
         return kmOut;
      };
      </script>
   </head>
   <body>
      <h1>JavaScript Function Example</h1>
      <form action="" onsubmit="formProcess()">
         <label>Enter miles </label>
         <input type="text" id="miles">
         <button type="submit">Submit</button>
   </body>
   </html>
```

This function picks up the value entered in the form text box.

It then passes the value to the milesConv() function and displays the result in an alert box.

This is the milesConv() function, which does the calculation.

The return statement sends the result back to the calling code.

This is where the main program calls the subprogram formProcess().

The submit button passes the user input into the subprogram.

In this code the milesConv function is called by the statement:

```
   let km = milesConv(theInput);
```

Where:

`milesConv` is the name of the function

`theInput` is a variable used to pass a value to the function (the number of miles to be converted)

`km` is a variable that is assigned the return value (the number of kilometres).

Object orientation

JavaScript is not a class-based object-orientated language (in contrast to Python), but you can use object-orientated programming techniques.

You can create objects, with properties and methods, with JavaScript in a similar way to how you create variables. Suppose you wanted to create an object to represent an order on an e-commerce website. The order object would have properties such as order number, order value, order date, and so on. These are defined as property-value pairs. For example:

```
   const order = {
      orderNum: 100,
      custName: "Evans",
      orderDate: "10th July",
      orderValue: 125.50,
   };
```

The value of each property can be accessed using: `order.orderNum`

You can declare methods for the object as part of the property definition for the object:

```
   function newFunc() {
      const order = {
      orderNum: 100,
      custName: "Evans",
      orderDate: "10th July",
      orderValue: 125.50,
      orderShipping: 10.00,
      calcTotal: function() {
         return order.orderValue + order.orderShipping;
         }
      };
```

A method called calcTotal is defined.

You can access the calcTotal function with `let total = order.calcTotal();`

EVENT TRIGGERS

You have already seen how JavaScript can respond to event triggers such as button clicks using the onsubmit() or onclick() methods. There are several other User Interface (UI) events that have methods that can trigger a function:

onload() – the onload event occurs when the page loads. This allows you to run a function as soon as the page loads into the browser.

oninput() – occurs when the user makes an entry in an input field, and avoids the need to include a button to process the user's input.

onchange() – this event occurs when an element changes. It is typically used with a select element in a form. When the user makes a selection from the drop-down list in the select element, the onchange() event occurs and a function can be run.

The following example demonstrates the use of the oninput() and onchange() methods. If you load this code in a browser it will also show the difference between them. With oninput() the function is triggered as soon as you type anything in the text box. With onchange() the function doesn't trigger until the text box loses focus. The oninput() method will not work with select elements. Note also that this example shows how you can pass an element ID to a function as a parameter, making the function able to work with several different elements.

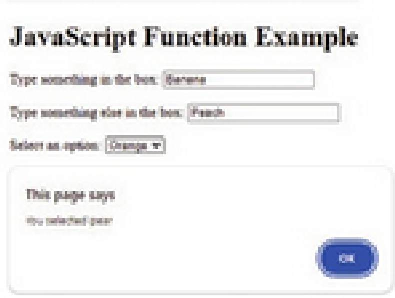
```
   <!DOCTYPE html>
   <html>
   <head>
      <script>
      function inputFunc(myIn){
         let theValue = document.getElementById(myIn).value;
         window.alert("You input " + theValue);
      }
      function selectFunc(){
         let theValue = document.getElementById("mySelect").value;
         window.alert("You selected " + theValue);
      }
      </script>
   </head>
   <body>
      <h1>JavaScript Function Example</h1>
         <label>Type something in the box: </label>
         <input type="text" id="myIn1" oninput="inputFunc(this.
          id)"><br><br>
         <label>Type something else in the box: </label>
         <input type="text" id="myIn2" onchange="inputFunc(this.
          id)"><br><br>
         <label>Select an option:</label>
         <select id="mySelect" onchange="selectFunc()">
            <option value="apple">Apple</option>
            <option value="orange">Orange</option>
            <option value="pear">Pear</option>
         </select>
   </body>
   </html>
```

The myIn parameter is used to pass the element ID to the function.

The element ID is passed to the function, 'this.id' meaning the element ID.

This Select element uses the onchange() event to run a function called 'selectFunc'.

There are two text boxes: one with the oninput() event set and the other with the onchange().


JavaScript Function Example

Type something in the box: Banana

Type something else in the box: Peach

Select an option: Orange

This page says

You selected pear

Keyboard events

The onkeydown event occurs when a user presses a keyboard key. This code example shows how this event could be used to change the colour of a text input box as soon as the user starts to type in it:

```
   <!DOCTYPE html>
   <html>
   <head>
      <script>
      function keyPress() {
         document.getElementById("change").style.backgroundColor = "red";
      }
      </script>
   </head>
   <body>
      <h1>Onkeydown event example</h1>
      <input type="text" id="change" onkeydown="keyPress()">
   </body>
   </html>
```

Using the ID of the text box, change the colour.

The onkeydown() event is added to the text box.

Mouse events

You have already seen the onclick event used to trigger a function when a user clicks a button. There are several other mouse events that can trigger a function, including:

oncontextmenu – right-click on an element

onmouseover – the mouse moves over an element

onmouseout – the mouse moves out of an element.

The following code shows an example of the use of the onmouseover and onmouseout events.

```
<!DOCTYPE html>
<html>
<head>
   <script>
   function mOver() {
   document.getElementById("mEvent").innerHTML = "Mouse Over";
   }
   function mOut() {
   document.getElementById("mEvent").innerHTML = "Mouse Out";
   }
   </script>
   <style>
   .testDiv {
      position: absolute;
      background-color: lightblue;
      top: 150px;
      left: 150px;
      height: 150px;
      width: 150Px;
   }
   </style>
</head>
<body>
   <h1>Mouse events</h1>
   <p id="mEvent"></p>
   <div class="testDiv" onmouseover="mOver()" onmouseout="mOut()">
     </div>
</body>
</html>
```

Functions that will be triggered by the mouse events.

Styling for a div that will be used to demonstrate the mouse events.

The onmouseover and onmouseout events are added to the div.

Focus and blur events

When an element receives **focus**, the **onfocus** event occurs; when focus moves away from an element, the **blur** or **onblur** event occurs. These events are often used with input elements on forms, and could, for example, provide contextualised prompts for users. The following HTML shows how this could be implemented using these events.

KEY TERMS

blur: when focus moves away from a particular element then a blur event occurs

focus: when an element receives focus it is ready to receive input from the keyboard or mouse

```
<!DOCTYPE html>
<html>
<head>
   <script>
   function onF(){
      document.getElementById("messF").innerHTML =
      "Product codes start with a letter followed by 3 digits";
   }
   function onB(){
      document.getElementById("messF").innerHTML = " ";
   }
   </script>
```

The onfocus function displays an information message.

The onblur function removes the message.

```
</head>
<body>
   <h1>JavaScript Focus Event Example</h1>
      <label>Enter product code: </label>
      <input type="text" id="prodCode" onfocus="onF()" onblur="onB()">
      <p id="messF"><p>
      <label>Enter other data: </label>
      <input type="text">
</body>
</html>
```

This text box has both onfocus and onblur events.

Form events

The most important form-related event is the submit event. This occurs when the user clicks a submit button or presses the enter within a form field. The function that is run when the submit event occurs is defined in the form tag; for example:

```
<form onsubmit="formFunction()">
   (HTML to create form elements)
<input type="submit">
```

The oninput() and onchange() events, which are often used with forms, were described earlier.

A summary of the events covered in this section is shown in Table 9.6.

TABLE 9.6: THE EVENT TRIGGERS

Event	Occurs when
onload()	the web page loads
oninput()	a user makes an entry in a field
onchange()	an element changes; used with select elements
onkeydown()	a user presses a key on the keyboard
oncontextmenu()	a user right clicks an element
onmouseover()	the mouse moves over an element
onmouseout()	the mouse moves away from an element
onfocus()	focus moves to an element
onblur()	focus moves away from an element
submit()	a user clicks the submit button on a form

SKILLS

ANALYSIS, PROBLEM SOLVING, ADAPTIVE LEARNING

ACTIVITY 3

1. What is the symbol for logical OR? What are the comparison operators for Equal to and Not equal to?
2. Suppose you are developing part of a website that sells train tickets. Passengers are given a discount of 20 per cent if they are under 18 years old or a 50 per cent discount if they are over 60 years old. Create a prototype page where the user enters their age and the page then displays the discount they are due.
3. Create an array with the names of some of the students in your class. Create a page that will allow you to add new names at the end of the array (use the push() method). Using the length() method, create a function to display all the names in the list.
4. Add functionality to the page you created above (using the onfocus and onblur events) that changes the formatting of the input boxes when the user types into them.

CREATING COMPONENTS USING HTML, CSS AND JAVASCRIPT

Combining the three technologies of HTML, CSS and JavaScript allows sophisticated components to be built. Some examples are shown here.

Creating a slideshow

A slideshow allows multiple images to be displayed one after another. You can create a very simple slideshow by changing the src attribute of an image when a user clicks the image. You need a way to keep track of the image being displayed so you can have the slideshow rotate around the available images. The following code shows an example of how this can be done.

```
   <!DOCTYPE html>
   <html>
   <head>
   <script>
      const myImages=["image0.jpg", "image1.jpg", "image2.jpg"];
      let currentSlide = 0;
      function slideFunc(){
         currentSlide++;
         if (currentSlide == 3){
         currentSlide = 0
         }
         document.getElementById("slideS").src = myImages[currentSlide];
      }
   </script>
   </head>
   <body>
      <h1>Image Slide Show Example</h1>
      <img src="image1.jpg" width=400px height=auto id="slideS"
       onclick="slideFunc()">
   </body>
   </html>
```

An array is created with the file names of all the images in the slideshow. A variable is also created for the current slide index number.

The slideFunc function increments the current slide and resets it to 0 if it reaches 3 (there are only 3 slides).

The value of the src property of the image tag is changed, using the array of image file names.

The onclick event is used to change the slides.

This version works fine but slideshow functionality is often implemented in a slightly different way. Rather than changing the src value, all the required <img> tags with the source file name listed are included in the HTML but their display property is set to hidden so they are not visible, and each slide is shown when required by changing the display property. Each of the <img> tags is given the same class name and this allows them to be referenced like an array. The following code example shows next and previous buttons to go forwards or backwards through the four different images.

```
   <!DOCTYPE html>
   <html>
   <head>
      <style>
      .theSlides {
         display: none;
         width: 50%;}
      .prevButton, .nextButton {
         position: absolute;
         top: 270px;
         padding: 10px;
         color: white;
         background-color: grey;
         font-weight: bold;}
```

Style for the slide initially sets the display property to none so the slides are not visible.

Styling for the next and previous links.

```
      .nextButton {
         right: 50%;}
      </style>
      <script>
         let slideNo = 0;
         let slideS = document.getElementsByClassName("theSlides");
         function firstFunc(){
            slideS[slideNo].style.display="block";
         }
         function slideNext(){
            slideS[slideNo].style.display="none";
            if (slideNo == 3){
               slideNo = 0;}
            else {
               slideNo++;}
            slideS[slideNo].style.display="block";
         }
         function slidePrev(){
            slideS[slideNo].style.display="none";
            if (slideNo == 0) {
               slideNo = 3;}
            else {
               slideNo--;}
            slideS[slideNo].style.display="block";
         }
      </script>
   </head>
   <body onload = firstFunc()>
      <h1>Image Slide Show Example</h1>
      <div>
      <img class="theSlides" src="clouds.jpg">
      <img class="theSlides" src="milkyw.jpg">
      <img class="theSlides" src="stars.jpg">
      <img class="theSlides" src="sun.jpg">
      <a class="nextButton" onclick="slideNext()">Next</a>
      <a class="prevButton" onclick="slidePrev()">Prev</a>
      </div>
   </body>
   </html>
```

The variable slideS points to the class to which all the images belong. This allows each image to be selected using `slideS[slideNo]`.

This function runs when the page loads and makes the first image visible.

The function displays the next slide by making the current slide not visible, incrementing the slide number and then making that slide visible.

This function shows the previous slide.

The onload event occurs when the page loads.

All the images are included and, as they all belong to the same class, they can be selected as if they are in an array.

This code uses a variable (slideNo) to keep track of the slide number. When the next or previous buttons are clicked, an if statement in the function checks if the slide number has reached 3 (for the next button, as there are only three slides) or 0 (for the previous button). If it does it resets the number (to 0 for the next button and 3 for the previous button), otherwise the slide number is incremented (for the next button) or decremented (for the previous button).

The technique of creating alternative page elements (images in this case) and then showing or hiding them by changing their display property can be used to implement a number of different web page features. Later in this section you will use the same technique to show or hide <div> sections of a page to implement tabbed content.

Modal boxes and images

A modal dialog box is one that appears in front of the current window. You can create a modal dialog by styling a div with CSS with its display property set to hidden, so initially it is not visible. The modal dialog can then be displayed by changing the display property of the div using JavaScript.

A modal image is a similar component to the dialog box, except the modal box contains an image.

Filter lists

In some applications you might want to filter a list based on some criteria. JavaScript provides a filter method that will filter the array based on a given criteria. This creates a new array containing only the matching array elements. The syntax of the filter method is a little complex. It looks like this:

Array.filter(function(*array item*){*return condition*;})

Array is the array containing the items you want to filter.

Array item is a reference to the current element in the array.

Condition is the condition used to filter the items in the array.

Let's look at an example of how the filter method can be used. Suppose you have an array of people's ages and you only wanted to list people over a given age, which is entered in a text input box. You could do this using the following code.

```
<!DOCTYPE HTML>
<html lang="en">
<head>
<title>Filter example</title>
<script>
const ages = [17, 23, 45, 21, 56, 19, 27, 19]
function filterAge(){
   let ageF = document.getElementById("ageIn").value;
   let newAge = ages.filter(function(arrayE){
      return arrayE > ageF;});
      alert (newAge);
   }
</script>
</head>
<body>
<h2>Filtering a list</h2>
<p>Enter your details to receive our monthly newsletter</p>
<form onsubmit = "filterAge()">
   <label>Enter min age </label><br><input type="text" id="ageIn"><br>
   <input type="submit" >
</form>
</body>
</html>
```

An array of various ages, called ages.

The function picks up the minimum age from the input box then uses the filter method to create a new array called newAge, with just the ages greater than the value the user has input. This is then displayed in an alert box.

The onsubmit event runs the filterAge function.

Sort lists

Sorting is a widely used data-processing task, which can be achieved using different sorting algorithms. The simplest is the bubble sort. A bubble sort involves repeatedly comparing adjacent values in an array and swapping them around if the left-hand value in the pair is larger than the right-hand value (for an ascending numerical sort).

Consider the values in this array:

4	3	7	9	2	6

The first two pairs of numbers, 4 and 3, are not in order, so they would be swapped, to become 3 and 4. The next two numbers will now be 4 and 7. These are in the correct order, so they are not swapped. The next two, 7 and 9, are also in the correct order. Next, you have 9 and 2, which are not in the correct order, so they are swapped to become 2 and 9. Finally, you have 9 and 6, which are not in the correct order, so are swapped to become 6 and 9.

So, at the end of the first pass through the array, you end up with:

3	4	7	2	6	9

The array is not fully sorted but the larger numbers are 'bubbling' to the far end of the array, hence the name bubble sort. In order to get the array fully sorted, it needs to be looped through again, swapping pairs that are out of order. This needs to be done repeatedly for a maximum number of times that is equal to the number of items in the array: six in this example.

The sort will work with both numeric and string data (with string data the sorting is done based on the character's ASCII values so they can be sorted in alphabetical order). To implement a bubble sort, two 'for' loops are used. The inner loop goes through the array once and swaps any adjacent items that are in the wrong order. The outer loop causes the inner loop to run as many times as there are items in the array. This will result in the list being full sorted. Here is a simple example of the bubble sort implemented in JavaScript:

```
<!DOCTYPE html>
<html lang="en">
<head>
   <meta charset="UTF-8">
      <script>
        const fruitArray=["pear", "banana", "orange", "apple"]
        function arraySort(){
         let temp="";
         alert (fruitArray);
         for (let y=0; y<5; y++){
            for (let i=0; i<5; i++){
               if (fruitArray[i] > fruitArray[i+1]){
                     temp = fruitArray[i+1];
                     fruitArray[i+1]=fruitArray[i];
                     fruitArray[i]=temp;
               }
            }
         }
         alert (fruitArray);
        }
      </script>
</head>
<body>
      <h1>Sort Example</h1>
      <button onclick="arraySort()">Sort</button>
</body>
</html>
```

This is the array to be sorted.

These are the inner and outer 'for' loops.

The inner 'for' loop uses an 'if' statement to compare two adjacent array items and swaps them around if they are out of order.

The function uses the alert method to display the array before and after sorting.

Pop-ups

You have already seen two types of pop-up: the alert box and the confirm box. Users must dismiss the pop-up (by clicking the OK button) before they can continue. The other type of pop-up is a confirm box. A confirm box is similar to an alert box but it contains OK and cancel buttons. The confirm() method returns true if the user clicks OK and false if they click cancel. The following code is an example of the use of the confirm() method.

```
<!DOCTYPE HTML>
<html lang="en">
<head>
<script>
   function userConfirm(){
      let message = "";
      let response = confirm("Are you sure?");
```

The userConfirm function uses a confirm() method to ask the user if they are sure. The return value from the confirm() method is assigned to the response variable.

```
      if (response == true){
         message = "Details are being processed";
      }
      else {
         message = "Processing cancelled";
      }
      document.getElementById("output").innerHTML = message;
}
</script>
</head>
<body>
<h1>Confirm example</h1>
<p>Content here could include a form the user fills in</p>
<button onclick = "userConfirm()">Process details</button>
<p id=output></p>
</body>
</html>
```

If the response is true then the user clicked 'ok' and an appropriate message is created.

If not then the user pressed 'cancel'.

The message is placed in the paragraph with the id of 'output'.

The onclick event for this button is used to trigger the function.

Bear in mind that although you can use the confirm button to allow a user to respond with 'ok' or 'cancel', you need to provide the code to implement these choices.

Tabbed content

Tabs allow a single page to display different content with a menu of tabs across the top. You can create tabs using CSS to style small clickable <div>s for each of the tab buttons and then content <div>s for each tab, which are hidden by setting the display property to none (rather like the images in the slideshow). Clicking on the tab buttons runs a JavaScript function that changes the display property so the content of the tab can be seen. The following code shows an example of this.

```
<!DOCTYPE html>
<html lang="en">
<head>
   <title>Tabbed content</title>
   <style>
      #tab1, #tab2, #tab3 {
         text-align: center;
         float: left;
         padding: 5px;
         margin: 5px;
         width: 75px;
         background: blue;
         color: white;
      }
      #tab1:hover, #tab2:hover, #tab3:hover {
         background: lightblue;
      }
      #contentT1, #contentT2, #contentT3 {
         width: 600px;
         margin-left: 5px;
         top: 40px;
         height: 150px;
         padding: 20px;
         border: 2px solid blue;
         position: absolute;}
   ##contentT2, #contentT3 {
         display:none;
   }
   </style>
```

Styling for the small tab buttons, which are positioned at the top of the page.

Change the button colour when the mouse hovers over them.

Styling for content tabs; these sit under the tab buttons.

Hide the second and third tab so when the page loads only the first tab is visible.

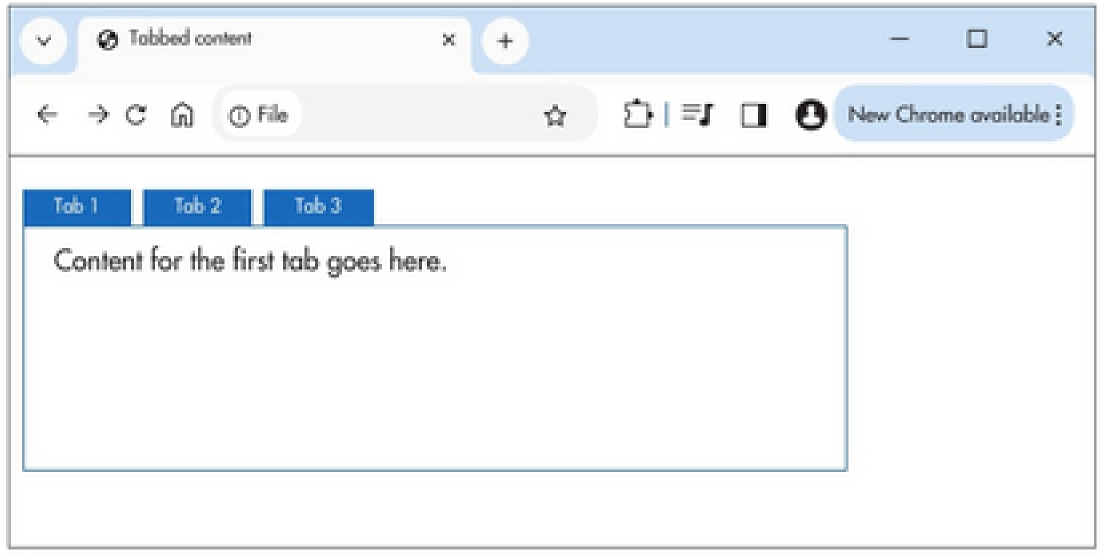
```
</head>
<body>
   <div id="tab1" onClick="showTab(1);">Tab 1</div>
   <div id="tab2" onClick="showTab(2);">Tab 2</div>
   <div id="tab3" onClick="showTab(3);">Tab 3</div>
   <div id="contentT1">Content for the first tab goes here.</div>
   <div id="contentT2">Content for the second tab</div>
   <div id="contentT3">Content for the third tab</div> <script>
   function showTab(tabNo) {
     document.getElementById("contentT1").style.display = "none";
     document.getElementById("contentT2").style.display = "none";
     document.getElementById("contentT3").style.display = "none";
     document.getElementById("contentT" + tabNo).style.display = "block";
}
   </script>
</body>
</html>
```

<div>s for the tab buttons; when clicked the showTab functions is passed the tab number.

<div>s for the tab contents

The showTab function is passed the tab number that the user clicked. You don't know which tab is currently visible so set them all to be hidden, then set the display style for the one the user clicked to block, so it is shown.

Figure 9.1 shows this code when displayed in a browser, with the first tab displayed.



Content for the first tab goes here.

▲ **Figure 9.1** A tabbed content page

ERROR HANDLING AND DEBUGGING

As mentioned earlier, debugging JavaScript is difficult because faulty code does not produce any error messages; it just does nothing. To help make developing and debugging code easier, use an incremental approach. This is particularly important when you need to develop a complex JavaScript function. Start by writing a few lines of code and check this works before adding a few more lines to the code and checking that works, before adding more functionality. If you write many lines of code all at once it can be very difficult to trace where the faults are.

Suggestions for resolving issues with code that does not work are as follows:

Try adding alert methods to trace the progress through the code. You can use alert methods to show the values in variables so you can check they are being used as expected.

The most common mistakes when writing code are:

- missing semi-colons at the end of a line
- non-matching brackets. If you use a programming editor like Notepad++ you can check brackets are in pairs by clicking on an opening bracket and the matching closing bracket will be highlighted in red
- capitalisation errors. JavaScript is case sensitive, so methods and properties must use the correct combinations of upper and lowercase.

If code doesn't work, you should try and narrow down where the problem is. You can comment out lines of code (by adding a double forward slash in front of the line) that you think might be causing the problem (which is quicker and easier than deleting them) to see if that enables the code to run.

Error handling

It is possible that errors will occur in the JavaScript code you write; for example a user may enter a value that is inappropriate for the field, such as text values where a numeric value is expected. You should therefore always validate user input where possible. You can also use what are called try–catch blocks, which will pick up any errors in the code within the try–catch blocks. The format of a try–catch block is as follows:

```
try {
    Code block
}
catch(error) {
    code to be executed if an error occurs
}
```

You can also add an optional 'throw' statement to set up specific error messages. An example of this is shown in the following code. One uses a function called NaN(), which means Not a Number, and returns true if the variable passed to it does not contain a number.

```
<!DOCTYPE HTML>
<html lang="en">
<head>
    <script>
        function myFunction(){
            let numb = prompt("Enter number");
            try {
                if(isNaN(numb)) throw "Not a number";
                if(numb < 0) throw "Number should be greater than 0";
                let acc=7/numb;
                document.getElementById("pOut").innerHTML = acc;
            }
            catch(err){
            document.getElementById("pOut").innerHTML = err;
            }
        }
    </script>
</head>
<body>
    <h1>Error handling</h1>
    <button onclick = "myFunction()">Click here</button>
    <p id="pOut"></p>
</body>
</html>
```

A try and catch block starts here because the user may have entered invalid data.

Two 'throw' statements are added; one uses the NaN function to check if the input is a number, the other checks it is not a negative number.

If an error occurs in the try block then the code in the catch block is executed. If it is one of the error set in the 'throw' statements, then this is what will be in the err parameter.

In this example, one error that could occur but is not covered by either of the 'throw' statements (although that could easily be changed) is if the user inputs a value of zero; this will cause a divide by zero error. This will be caught by the try and catch block but the default error message (which for this error is 'infinity') will be assigned to the err parameter.

There are free websites where you can test and debug HTML, CSS and JavaScript; for example, search online for JSFiddle.

SKILLS

ANALYSIS, PROBLEM SOLVING, ADAPTIVE LEARNING

ACTIVITY 4

1. There are several methods you can use to provide a slideshow. Describe one of them.
2. What does the term 'modal' mean in relation to web page windows?
3. Outline the steps used in a bubble sort.
4. List two common errors when typing JavaScript code.
5. How is a try and catch code block created?
6. Create a type of slideshow that uses tabbed content. Use at least four different images, each one shown on a different tab. Usually you swap tabs by clicking the tab at the top but you could also add buttons on each tab to take you to the previous or next tab. You could add further functionality by creating a box that pops up, describing the image when it is clicked or when the mouse moves over it.

CHECKPOINT

1. Explain what an API is.
2. Where in an HTML document can you insert JavaScript code?
3. How can you define a variable in JavaScript?
4. What is the difference between a one-dimensional and two-dimensional array?
5. What is a regex and how can it be used?
6. What are the onfocus and onblur events and when do they occur?
7. How are comments added to JavaScript code?
8. What is the difference between a 'do' loop and a 'do while' loop?
9. Give two examples of mouse events.
10. What must you place at the end of each JavaScript statement?

THINKING LIKE AN IT SPECIALIST

JavaScript is a very widely used programming language, which means there are a lot of materials available online and a large number of code examples. Therefore, if you want to write JavaScript to carry out some tasks on a web page it is likely that you can find a similar if not identical example of that code by searching online. There are, of course, both advantages and disadvantages to using code that's been written by someone else. What do you think those advantages and disadvantages might be?

EXAM TIP

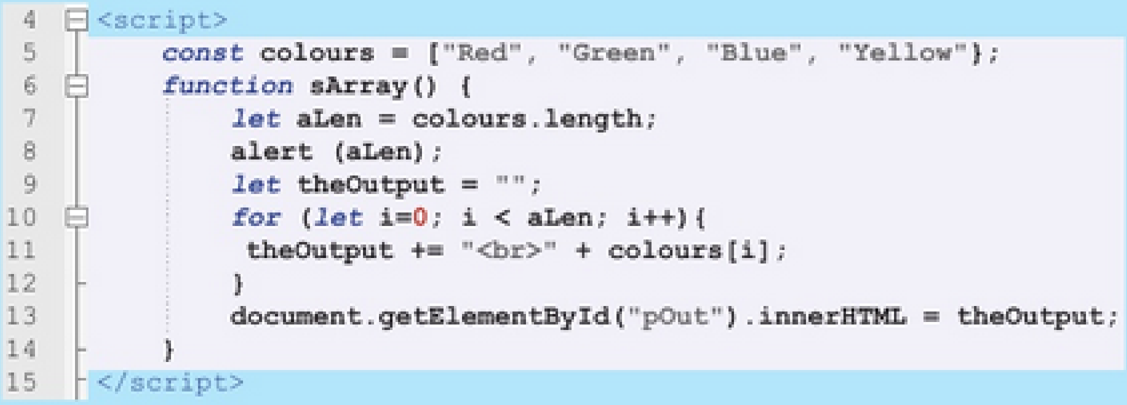
Section A of the exam typically has three or four questions on JavaScript at the end of the section (after the questions on HTML and CSS). You might be given a small piece of JavaScript code and be asked to describe what it does. Or you might be provided with some JavaScript code that is incomplete and asked to complete it or add further functionality. You may be asked to write, explain or amend a regex expression. Section B of the exam paper typically asks you to create a web page to a provided design using HTML and CSS. You may be asked to add functionality to the web page you create using JavaScript. You should remember to add comments to any JavaScript code you write in the exam.

SKILLS

INTERPRETATION, PROBLEM SOLVING, CRITICAL THINKING

EXAM PRACTICE

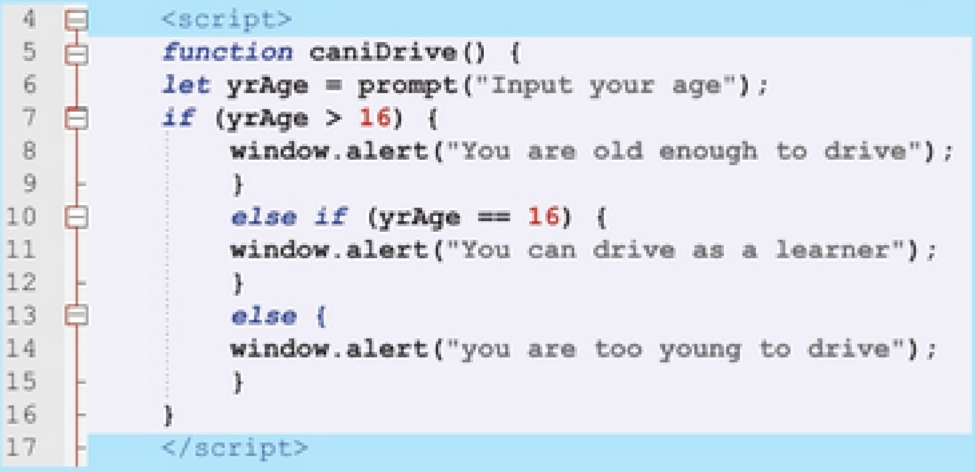
1. Give one benefit of using indentation when writing JavaScript code. [1 mark]
2. JavaScript provides interactivity for web pages.
   - **a.** Give one feature of the DOM (Document Object Model). [1 mark]
   - **b.** Describe what the 'onload' event is and how it can be used. [2 marks]
   - **c.** The code shown here contains a 'for' loop.



```
4   <script>
5       const colours = ["Red", "Green", "Blue", "Yellow"};
6       function sArray() {
7           let aLen = colours.length;
8           alert (aLen);
9           let theOutput = "";
10          for (let i=0; i < aLen; i++){
11           theOutput += "<br>" + colours[i];
12          }
13          document.getElementById("pOut").innerHTML = theOutput;
14      }
15  </script>
```

Describe how the code works. Use line numbers to support your description. [4 marks]

3. This image shows the JavaScript used to decide if someone is old enough to drive:



```
4        <script>
5        function caniDrive() {
6        let yrAge = prompt("Input your age");
7        if (yrAge > 16) {
8            window.alert("You are old enough to drive");
9            }
10           else if (yrAge == 16) {
11           window.alert("You can drive as a learner");
12           }
13           else {
14           window.alert("you are too young to drive");
15           }
16       }
17       </script>
```

Describe how the code works. Use line numbers to support your description. [4 marks]

4. Regular expression can be used to validate user input.

This is an example of a regular expression:

d{4}[a-z]{2}(x|y)[A-Z]E+

Give two reasons why the input 123feysEE would fail the validation. [2 marks]

SELECT FILE.
COVER
WORDS
PLAY LIST
WEB DESIGN
Effect
–Blure
–Speed
OUTPUT RESOURCES
3 PIC
RESEARE

TOPIC 10 DESIGNING WEB PAGES

Designing and developing web pages is a very large industry that is constantly evolving. There are some fundamental design principles that are important to follow when designing and developing web pages. These principles exist to make sure that all web pages are accessible for every user, whatever their specific needs. In this topic you will learn about these principles of design and what can be done to make sure a website is accessible to everyone.

TOPIC 10 DESIGNING WEB PAGES

LEARNING OBJECTIVES

By the end of this topic you should be able to understand:

- how design tools such as wireframes, mock-ups, style guides and prototyping are used to design web pages
- what web-safe fonts are
- how to use design principles to create effective page layouts and designs
- how to make web pages accessible to everyone
- how to design helpful navigation systems
- how to target specific devices and web browsers.

GETTING STARTED

Look at a website that you use regularly. Consider the following questions:

- What do you notice about the design?
- What kind of colour scheme is used?
- Where are the navigation elements of the website?
- What features do you find easy to use on the website?
- What features do you find difficult to use on the website?
- Would you rate the design of the website as unsatisfactory, satisfactory, good or excellent?

▲ A web designer exploring the use of colour charts in their work

PRINCIPLES OF DESIGN

The design of web pages has changed greatly since the start of the internet. The first web page went live on the internet in 1991 and contained information about the World Wide Web project. This web page is still available in its original form on the Cern website (search online for info.cern.ch).

Early web pages were mainly text based with some low-quality images. This was due to the low bandwidth that was available to transmit data across the internet. As web design developed, web pages became more advanced and began to include many different elements such as animations, videos and interactive elements.

The fundamental principles of web design have remained fairly similar for a long time now. The most important element of web design is to keep the design consistent. This involves consistent use of colours and fonts. You will learn more about the use of colours and fonts later in this topic.

HOW TO DESIGN WEB PAGES

Before a web page is built, you will need to create a design for the page. There are several tools that can be used to create the design, including:

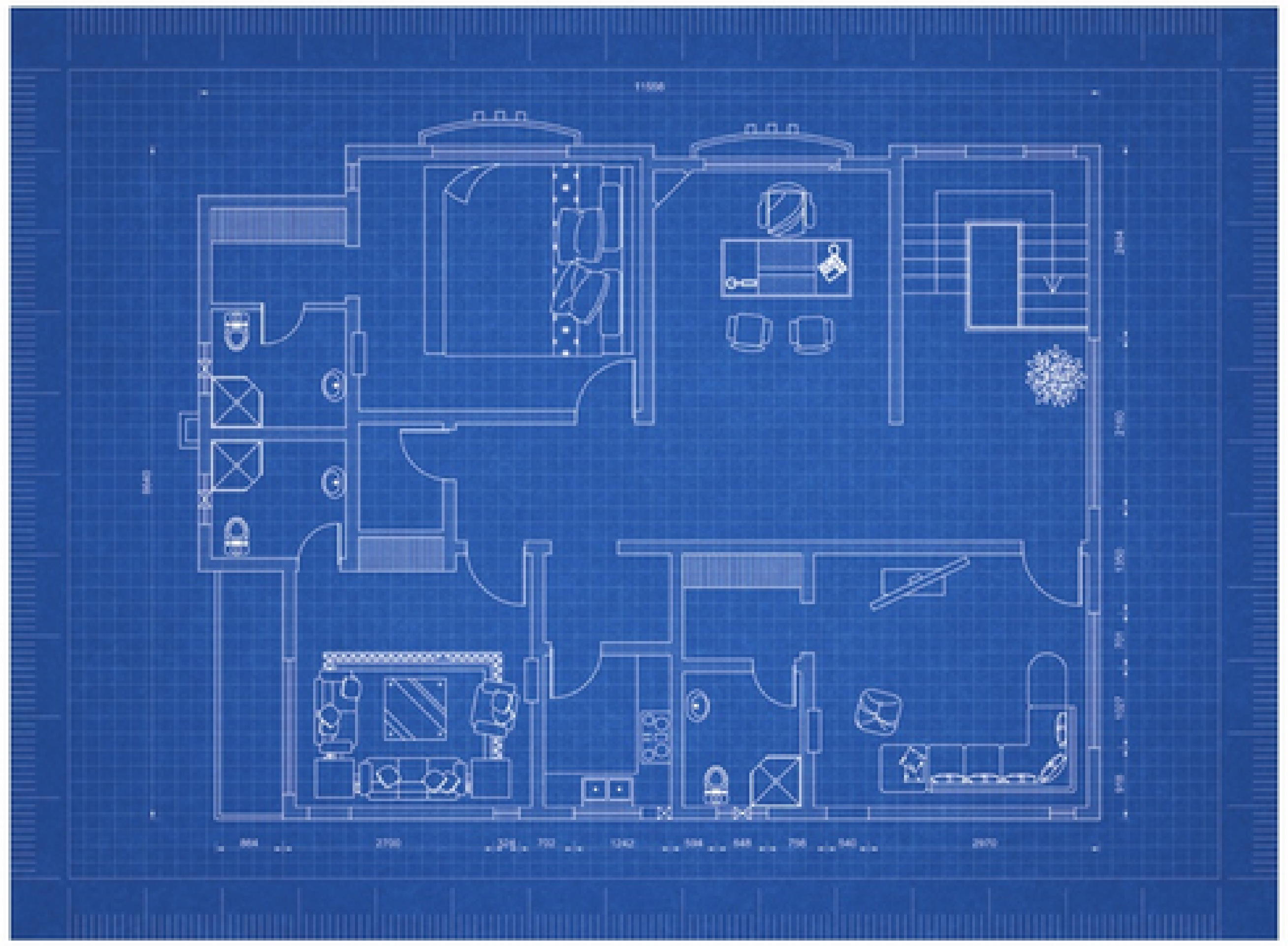
- wireframes
- mock-ups
- style guides
- prototyping.

Wireframes

A **wireframe** is like a 2D map of the web page. It shows how the different content should be laid out for the web page. It is similar to a blueprint for a building.

KEY TERM

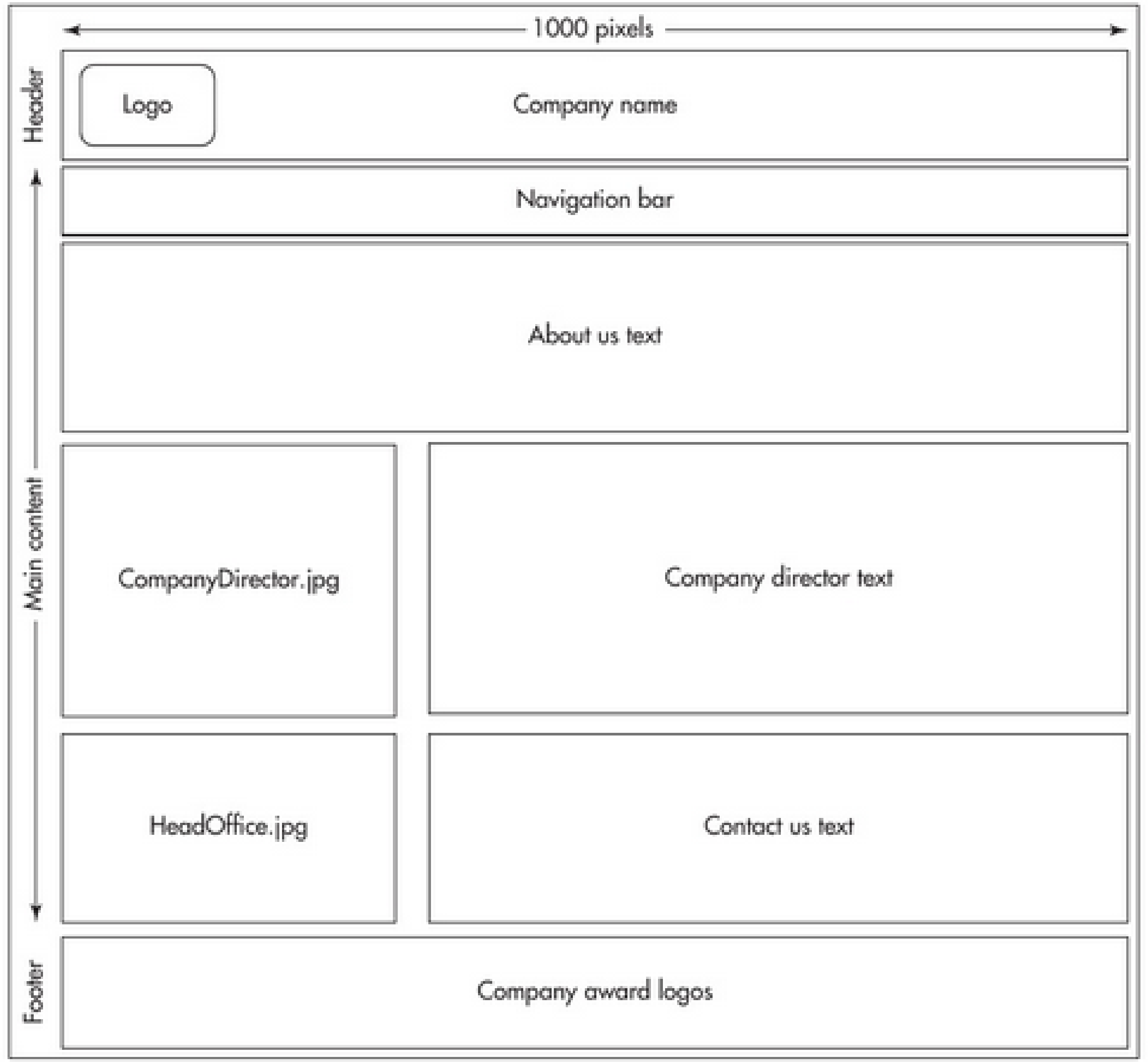
wireframe: a 2D blueprint of the layout of a web page



▲ A blueprint for a building shows the layout

A wireframe is used to show the basic structure of the web page. It specifically focuses on the layout of the page, but not the style or content of it. Those aspects are defined in other design tools.

Figure 10.1 shows how a simple wireframe for a web page could look.



▲ Figure 10.1 A simple wireframe for a web page

When you look at this wireframe you can gain a lot of information about where all the content that will be included in the web page will be placed. You can see what will be placed in the header, main content area and the footer. You can see where all the text and images will be placed. You can also see where the navigation bar will appear. The information that you do not gain from looking at this wireframe is what colours will be used on the page, what font will be used for the text and what kind of navigation system will be used.

SKILLS
CREATIVITY

ACTIVITY 1

Think about your favourite hobby or activity. You are going to design a web page that tells people about you and your favourite hobby. Create a wireframe to show the layout of your web page.

KEY TERM

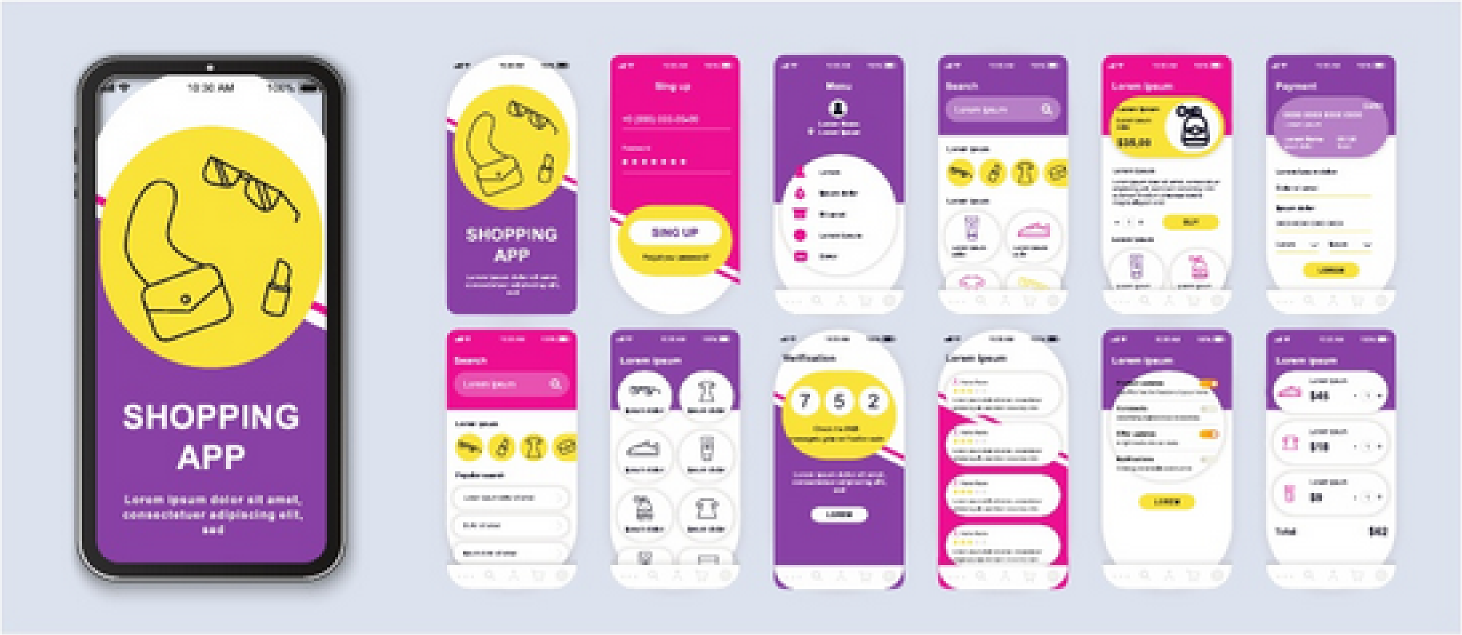
mock-up: a realistic representation of what a web page will look like

Mock-ups

A **mock-up** is similar to a wireframe in that you can also see what the layout of the web page will be. However, a mock-up is a more realistic representation of what the final web page will look like when it is built. It normally has the actual text that will appear in each allocated section, if the text is currently known. It can also often have the actual images that will appear. The benefit of creating a mock-up for a web page is that the designers and developers can see a more accurate visual representation of what the final product will be. They can see if the chosen fonts work well and if the colours create the desired effect.

A mock-up is usually created digitally and will include styling elements for the web page, such as the colours and the fonts that will be used.

Figure 10.2 shows how a mock-up for a shopping application for a mobile device could look.



▶ Figure 10.2 A mock-up of a shopping application for a mobile device

SKILLS
CREATIVITY

ACTIVITY 2

Use the wireframe that you created in Activity 1 to create a mock-up of your web page.

KEY TERM

style guide: a document that defines all the style formatting elements for a website

Style guides

A **style guide** for a web page shows more in-depth detail about how the web page will be formatted. The style guide is like the rules for creating the web page. It includes information such as:

- the hierarchy of text that should appear, e.g. heading, subheading, main text
- the font style, size and colour of each heading, subheading and main text
- instructions about where to use special text formatting such as bold and italics
- how items such as tables and lists should be formatted
- how buttons and icons should be formatted
- guidelines about how images should be displayed.

The reason a style guide is created for a web page is to make sure that the styling for each web page in a website is consistent. Most companies want the style of their web pages to be consistent because it makes them appear more professional. Any web page that is added to the website should always be formatted using the style guide. The benefit of a company having a style guide is that any web developer will easily be able to add another web page to the website if necessary. It is not limited to the person who first built the website.

An example of part of a style guide for a website could be:

Heading 1

Font colour: #000000

Font style: Arial

Font size: 18px

Subheading

Font colour: #000000

Font style: Arial

Font size: 14px

Main text

Font colour: #000000

Font style: Arial

Font size: 12px

SKILLS

INTERPRETATION

ACTIVITY 3

Create a style guide for your web page that could be used by another developer to easily add another page to your website.

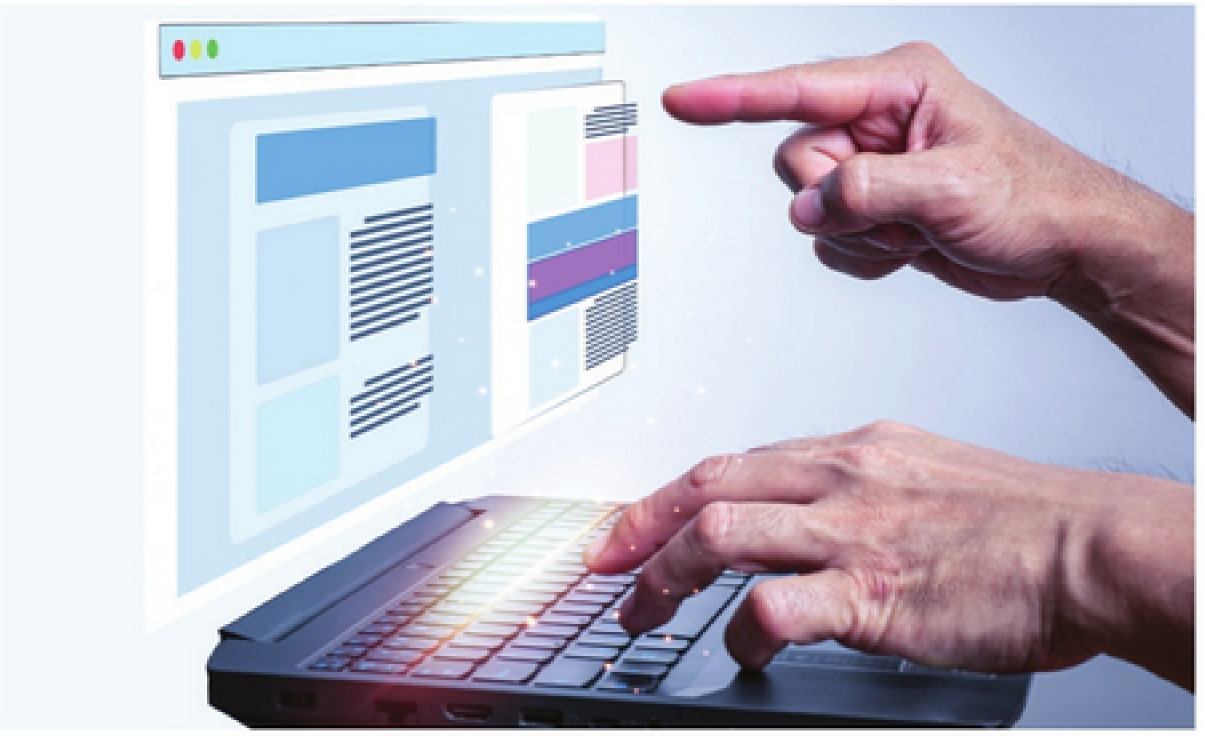
KEY TERM

prototype: a working digital model of what a website will look like

Prototyping

A **prototype** for a website is a digital model of what a web page will look like before the whole website is built. It is like the first draft of the final website that will be built. It allows a user to see how the website will function and how the user will interact with the website.

A prototype of a website page is often built to allow a person to use the website and give feedback on using it. This can allow the website designers and developers to implement any changes that they feel would improve the website before the final version is built. The benefit of this is that it avoids time being wasted creating a full website that may need significant changes because of feedback at a later stage.



▶ A prototype allows a user to interact with a web page and provide feedback on using it

WEB TYPOGRAPHY

You need to consider the style of the fonts that are included in a style guide. A designer should make sure that the fonts chosen are web-safe fonts. A web-safe font is a standard font that can be used by any device because it is pre-installed on the device. It means that the font will display correctly and the user will not need to download any special fonts. Some common web-safe fonts are:

Arial

Georgia

Courier New

Impact

Tahoma

Times New Roman

Verdana.

If you have a strong reason to use a different kind of font, one that is not a standard web-safe font, there is a process to do this. This is called embedding a font into the web page. When you create the web page, you can embed a small file into the page that contains the data needed to display the special font. When the user downloads the web page, the file is also downloaded and the data in it is used to display the web page with the special font. For example:

```
   <link rel="stylesheet"
         href="https://fonts.googleapis.com/css?family=Asap">
      <style>
      body {
      font-family: 'Asap';
      font-size: 12px;
      }
      </style>
```

EXAM TIP

In the exam you may be given a wireframe and a style guide to create a mock-up or a prototype of a website. The detail given in both is important and will be carefully assessed. You should double-check all the formatting that you show in your answer and make sure it matches the wireframe and style guide.

EFFECTIVE DESIGN

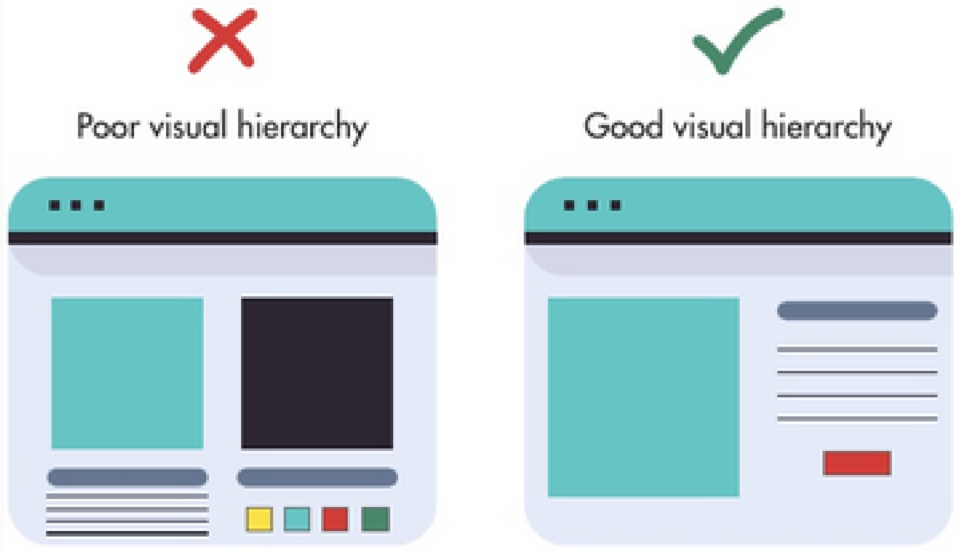
Design principles are guidelines that can be used when designing a web page. They exist to make it easier for the designer to understand how to create a web page that is suitable for a large number of users. The design principles are based upon science and years of tried-and-tested methods that have been used when designing and developing web pages. They are like a wise old leader who can pass on their wisdom about what makes the design of a web page good. The design principles are based around key areas. Table 10.1 shows the key areas with a description of each.

KEY TERMS

flow: the path a designer creates through a website for a user to follow

visual hierarchy: using different formatting on items to make them stand out more than others

TABLE 10.1: THE KEY AREAS OF DESIGN PRINCIPLES

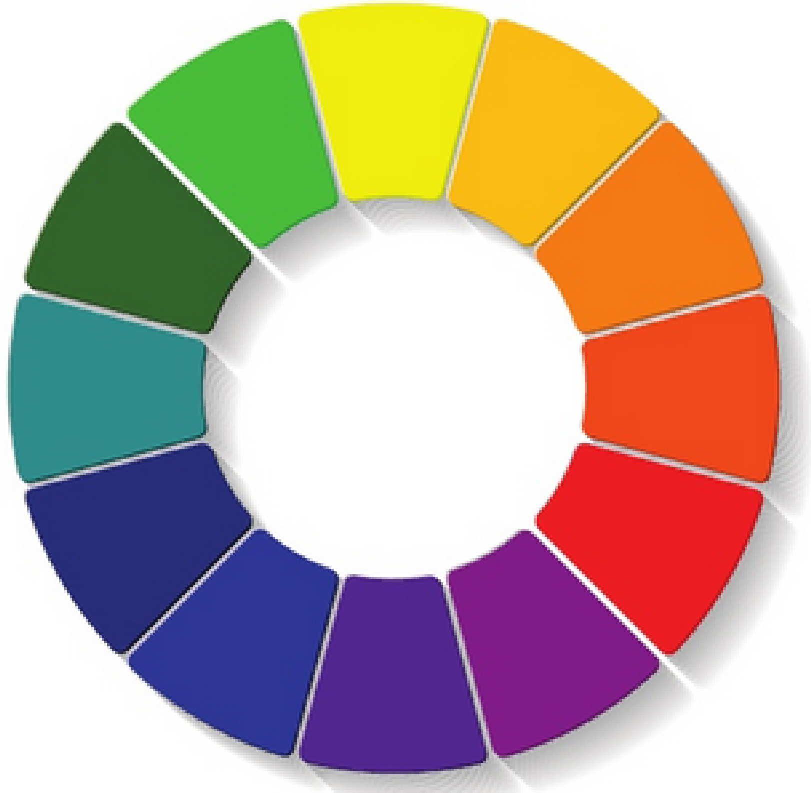
Key area	Description
Visual hierarchy	This refers to the importance of the elements that appear within the design. When the web page is displayed it should be obvious what the more important elements on the page are. This is because they should stand out from the rest of the page. If you open a web page and have a hard time seeing where to look, the visual hierarchy of the web page is likely to be poor. The designer has not made the important elements stand out enough. If you open a web page and are instantly drawn to certain parts of it, it is likely the designer has decided that they are the more important things that they want you to view first. A designer may create visual hierarchy by using: • size – a user will probably notice larger items • colour – a user may be more attracted to brighter colours or colours that contrast • alignment – items that are not in a usual alignment may stand out more to a user because they are different • proximity – items placed higher up on a page or closer to another important item may be seen first • white space – more space around an item may mean it is more noticeable to a user.  Poor visual hierarchy Good visual hierarchy ▲ Different elements combine to create a good visual hierarchy
Flow	The flow is the path that the designer has created for the user to follow through your website. If the user is completing a task with the website, such as buying a product or service, this is the flow of the actions that they carry out to do this. The designer should think carefully about whether each action that the user should be completing is obvious. For example, is it obvious where the user can find the virtual shopping cart when they have added all the items they want? Is it obvious where the user should choose the correct colour and size for the item that they want? Are these choices carried out in an expected order? For example, you wouldn't expect to be able to add a sweater to your shopping cart without first choosing the size that you want. You created a dataflow diagram for an online shop in Topic 4, Activity 4.

(*Continued*)

KEY TERM

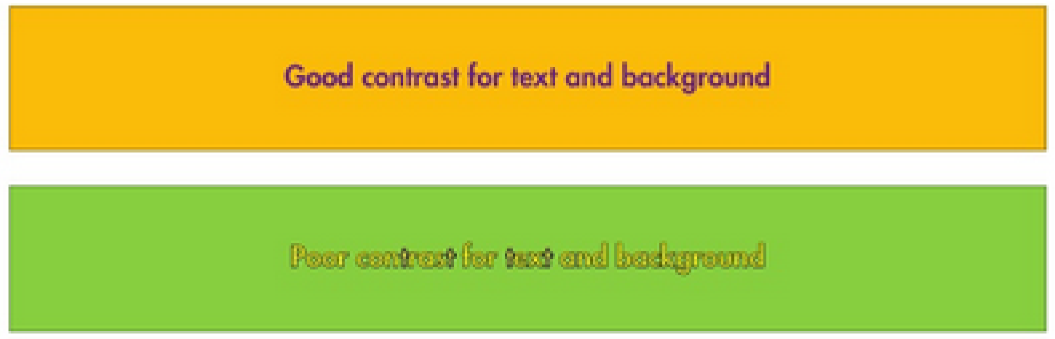
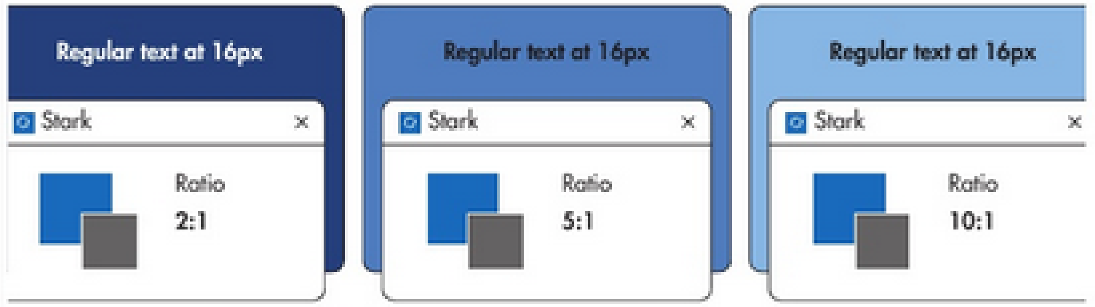
colour theory: the understanding of how colours work together and how they compare and contrast

TABLE 10.1: THE KEY AREAS OF DESIGN PRINCIPLES (*Continued*)

Key area	Description
Colour theory	This is the understanding of how colours work. It involves knowing how colours work together, such as how they complement or contrast with each other. It also involves understanding the effect certain colours can have on our emotions and perceptions. To understand colours that complement each other, colours that contrast and colours that really do not work together, a designer can use a colour wheel. The colour wheel was created by Sir Isaac Newton. You may associate him more with his developments in physics, but he also created a wheel that shows the relationship of colours to each other. This is a basic example of a colour wheel:  ▲ A simple colour wheel Colours that are opposite each other in the wheel are known as complementary colours. For example, yellow and purple are complementary colours. Complementary colours are used when a sharp contrast needs to be created. This is because the two colours are very different and can make an item stand out. Colours that are side-by-side in the wheel are known as analogous colours. For example, red, mid-purple and dark purple are analogous colours. Analogous colours are used because they blend well together. These colours are reasonably similar and can create a pleasing effect when used together. There are normally three colours used in an analogous colour scheme. Three colours that are evenly spaced out around the colour wheel are called triadic colours. For example, yellow, blue (mid) and red are triadic colours. Triadic colours are used in a similar way to complementary colours. They are all quite different and contrast well to make items stand out.

(*Continued*)

TABLE 10.1: THE KEY AREAS OF DESIGN PRINCIPLES (*Continued*)

Key area	Description
	We can also attach different emotions and perceptions to colours. For example, the colour red may mean danger to some people but to others it may mean good fortune. Yellow is often seen as a happy colour and green a more calming colour. If a designer wants to create a certain feeling for the user about their website, they may choose to use colours that are known to be associated with this kind of feeling. The use of the colour wheel to choose colours for a web page can help to create both balance and contrast for the web page. Brightness and contrast in general are two important factors when choosing colour for a website. If a background colour and text colour are chosen that do not contrast well, it may make it very difficult to see the text. For example:  Good contrast for text and background Poor contrast for text and background ▲ Background colours need to contrast with text colours for good readability For brightness, the **World Wide Web Consortium (W3C)** recommends a minimum brightness and contrast ratio of 4.5:1 for standard text and 3:1 for larger text. This is to make sure that text can clearly be seen but to also make sure that users do not get eye strain from colour contrasts that are too bright. Here are three different ratios of black text on a coloured background:  Regular text at 16px Stark Ratio 2:1 Regular text at 16px Stark Ratio 5:1 Regular text at 16px Stark Ratio 10:1 ▲ Different brightness and contrast ratios can be used

KEY TERM

World Wide Web Consortium (W3C): an organisation that sets standards for the world wide web

SKILLS

ANALYSIS, CRITICAL THINKING

ACTIVITY 4

Look at the design you created for your web page. Focus on how you have used colour and see how that compares to the colour wheel and colour theory. Are you happy with the colours that you have chosen? Do you want to change any? Make any changes that you feel would improve the use of colour in your web page design.

KEY TERMS

disabilities: conditions of the body or mind that make it more difficult for a person to carry out certain actions

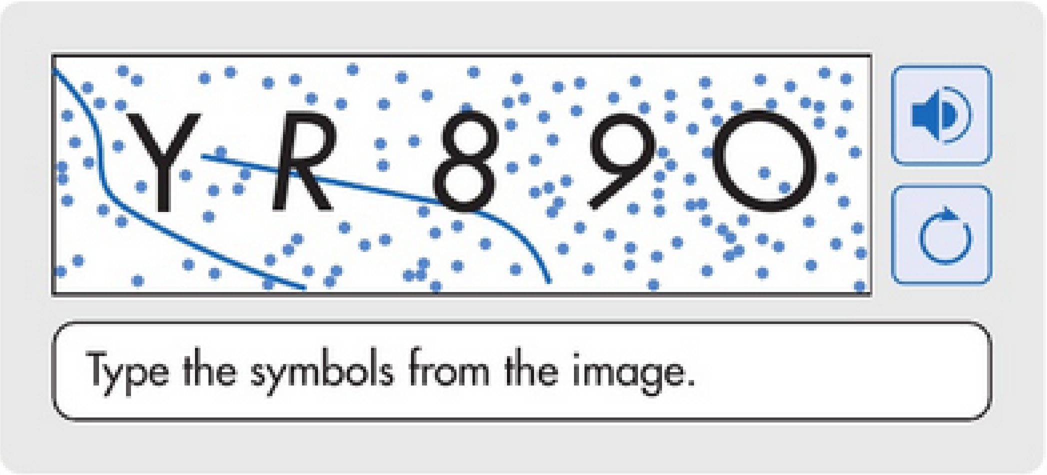
Web Content Accessibility Guidelines (WCAG): guidelines that set standards for what should and should not be included in websites to make them more accessible

PRINCIPLES OF ACCESSIBILITY

When you create a website, it is likely to be used by a wide variety of people. People are all different and use things in their own individual ways. Some people using the website will also have **disabilities**. Therefore, you may want to include a wide range of features that can help the website to be accessible to everyone that wants and needs to use it. Making a website more accessible to people means making it as simple as possible for everyone to use.

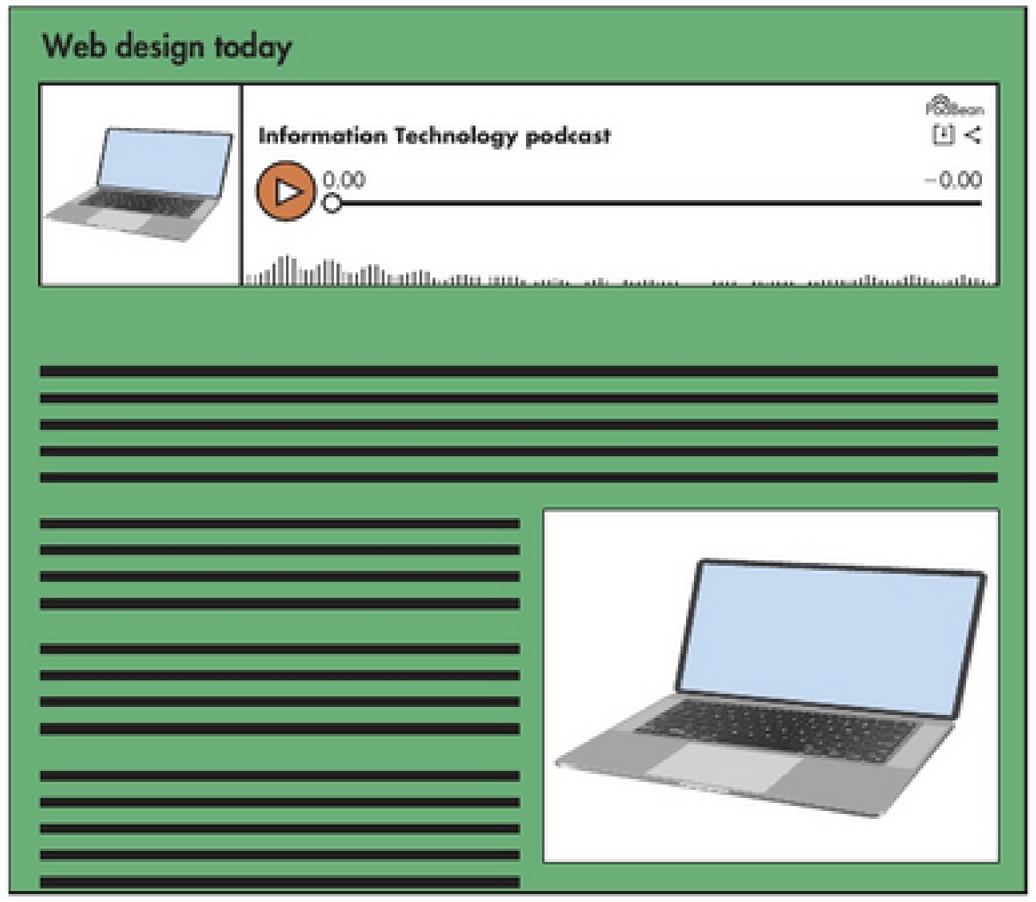
There are some guidelines that companies can use to help design their website to make it more accessible for those with disabilities. These are the **Web Content Accessibility Guidelines (WCAG)**. The current version of the guidelines was released in 2001. There is a new set of guidelines under development that should be released in the future. Table 10.2 shows the areas that are covered by the current WCAG and some information about the application of these areas.

TABLE 10.2: THE WEB CONTENT ACCESSIBILITY GUIDELINES

Guideline area	Description and application
Text alternatives	This refers to providing text alternatives for non-text items in the web page. The way this can be applied is by providing a description of an image. This means that users who may have limited or no sight can still read the description of the image and know what it is. They often do this by using text to speech software. You can also provide text alternatives for any test methods that are used when logging into a website, such as CAPTCHA. CAPTCHA can often require elements to be identified in an image, so the text description is important to assist with this. For example:  ▲ An example CAPTCHA In this example, a user can see the text that they need to enter but they can also hear a text alternative by clicking the sound icon on the right. If they click this icon, the text will be read out to them.

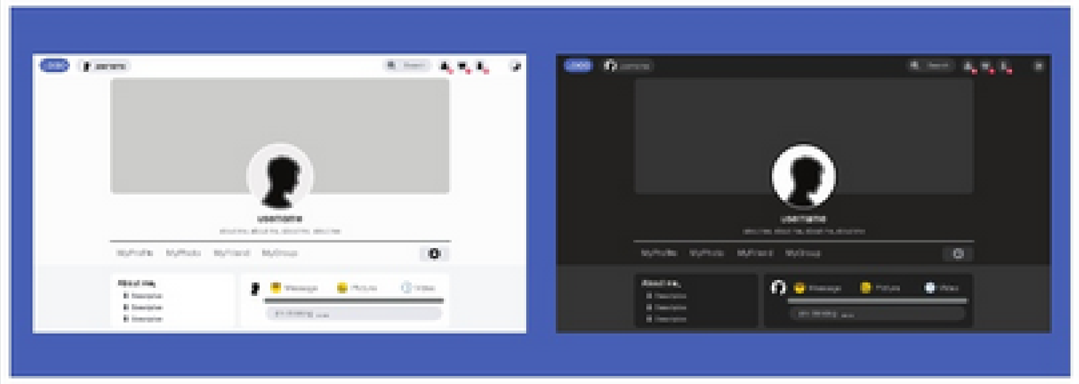
(*Continued*)

TABLE 10.2: THE WEB CONTENT ACCESSIBILITY GUIDELINES (*Continued*)

Guideline area	Description and application
Time-based media	Time-based media in a website is anything that is animated, makes a noise or changes without the user's input. A website should provide an alternative for any time-based media that is not accessible to everyone. For example, if the website has an audio-only recording, it should provide a written description of what is included in the recording. This allows users who have limited or no hearing to be able to understand any information the recording may contain. Also, if the website contains a video that has an audio commentary, it should be possible for the user to see closed captions and subtitles for the video. This helps users who cannot hear the video. It can also be helpful for a website to include a person signing any content that may not be pre-recorded. This will help those who are unable to hear any content. For example:  Web design today Information Technology podcast 0.00 −0.00 PodBean This web page has a podcast section at the top and a text-based transcript underneath
Adaptable	The content of the website should be adaptable. This means that it should be possible for it to be displayed in different ways depending on the user's needs. It should be able to adapt to any orientation needed by the user's device. If a user has sensory issues, meaning that they may find certain colours and sounds too much, they should be able to have a reduced sensory version of a website that doesn't have lots of loud and colourful features.

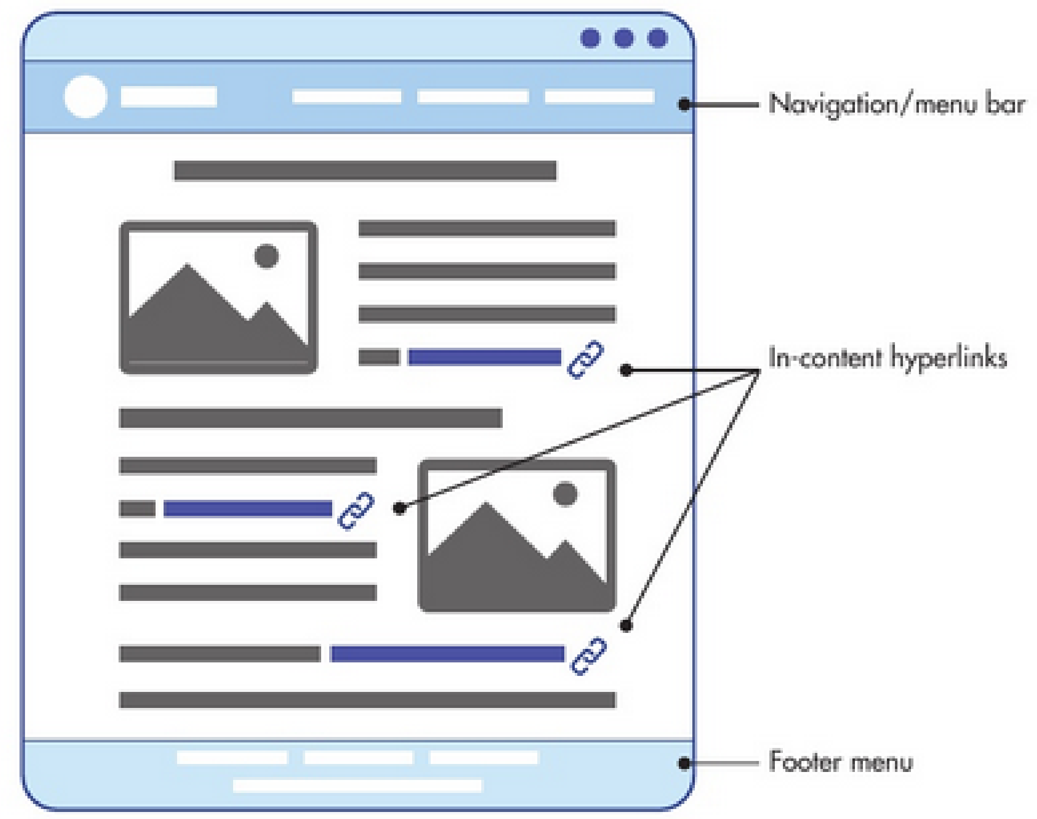
(*Continued*)

TABLE 10.2: THE WEB CONTENT ACCESSIBILITY GUIDELINES (*Continued*)

Guideline area	Description and application
Distinguishable	The content of the website should be distinguishable. It should be easy for everyone to see and hear the content of the website. For example, one feature that is used by some users to view the content on a website more easily is dark mode. This switches any text that may be written in black or a dark colour on a white background, to text written in white or a light colour on a dark (often dark grey/black) background. This can help some users see the text more clearly. It is often helpful to users who have disabilities such as dyslexia.  Standard mode versus dark mode Another feature that can be used in a website to make it more accessible in this way is the ability to increase the size of the text. This can be useful for those using the website, especially older people, as it can make the text more distinguishable and easier to read.
Keyboard accessibility	This simply means that all features of the website should be accessible using a keyboard. A user should be able to navigate to and select any item on the web page using only a keyboard.
Enough time	Where any content is moving, changed or updated while a user is using the website, they must have enough time to read/view the content. For example, if a website has a scrolling image gallery, the user must be given enough time to look at each image in the gallery. Websites are encouraged to allow the ability for a user to turn off any time restrictions in moving or changing content. They are also encouraged to allow a user to override any timed content with manual inputs, such as a mouse click.
Seizures	Websites should consider content for anyone who may suffer seizures. They should not provide content that flashes more than three times in one second. This type of content is known to trigger seizures in those that suffer from them.

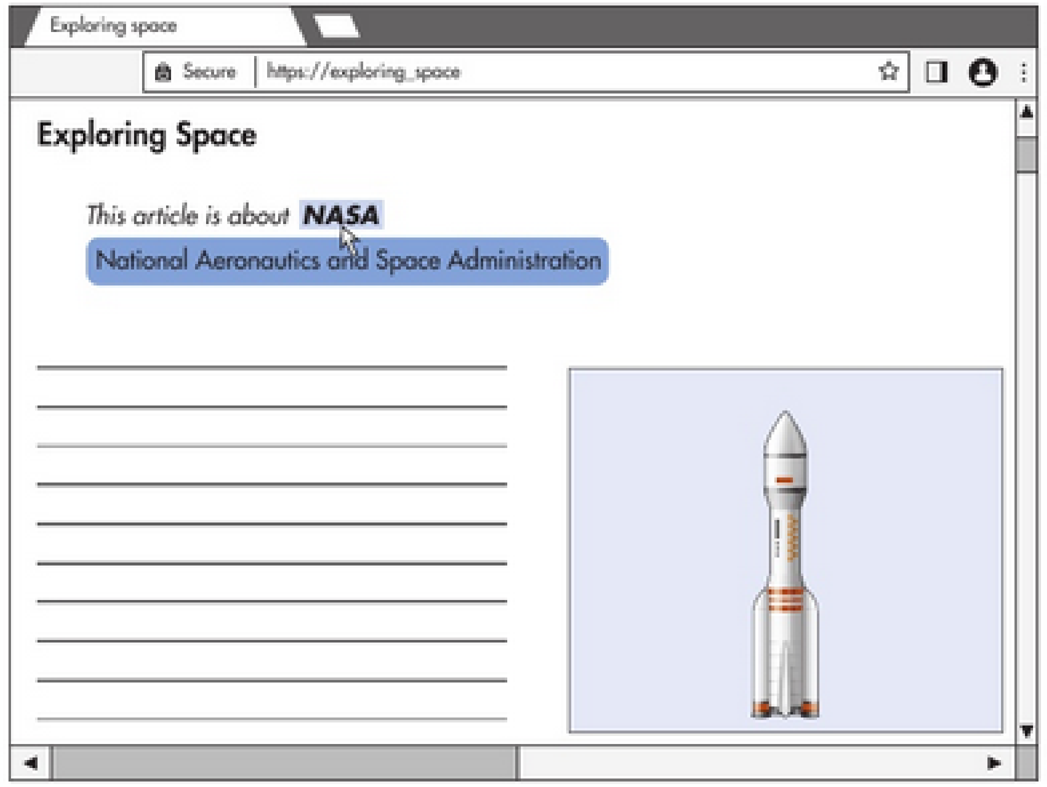
(*Continued*)

TABLE 10.2: THE WEB CONTENT ACCESSIBILITY GUIDELINES (*Continued*)

Guideline area	Description and application
Navigable	A website should have lots of features that allow a user to easily navigate it. These can include a menu bar, hyperlinks, and clear and precise titles for areas of content.  ▲ A website needs good navigability Websites should also include features such as bypass blocks. This is a feature that allows a user to quickly skip past a section of content.
Input modalities	These are features that make it easier for a user to interact with the website beyond the use of a keyboard; for example, by clicking content with a mouse or on a touchscreen. Websites may also choose to include features that allow motion as a method of input. For example, if a website has a game that requires a user to get a marble from one side of a maze to another, they can physically move and tilt their device to provide the input for which direction the marble should roll. One feature that websites have to be cautious in using is the dragging of content from one part of the website to another. This can sometimes be a difficult action for people, especially those with physical disabilities that affect the control of hands or fingers. A website should make it possible to move something from one place to another using a single click instead of a dragging action.

(*Continued*)

TABLE 10.2: THE WEB CONTENT ACCESSIBILITY GUIDELINES (*Continued*)

Guideline area	Description and application
Readable	The language used on the website should be carefully considered. Any technical jargon should be explained and any acronyms should be written out in full at least once. A website may choose not to explain all jargon within the text but may include features such as hovering over the word so that a description appears. For example:  Exploring Space This article is about NASA National Aeronautics and Space Administration ▲ Technical jargon and acronyms must be explained Websites are also encouraged to include the pronunciation of any words that may be difficult to pronounce or unusual.
Predictable	We all have mental models of how things work. These models are developed in our minds through repeating actions many times. A website should make sure that any content is in a predictable place. For example, you would mostly expect a navigation bar for a website to be across the top of a page or down the left-hand side of the page. If a website has its navigation bar across the bottom of the page, this could cause confusion for users as they will not find it in a predictable place. Any common content on the different web pages of a website should also appear in the same place on each page. For example, the navigation bar should be in the same place on each web page.

(*Continued*)

TABLE 10.2: THE WEB CONTENT ACCESSIBILITY GUIDELINES (*Continued*)

Guideline area	Description and application
Input assistance	Input assistance helps users to complete and correct any inputs they make when interacting with the website. An example of this would be when a user is filling out their details on a web page; if they input data that does not match the rules needed for that data, the user is immediately notified. For example, if they input their telephone number but it is longer than the required number of digits, they are told the telephone number is too long. Users should also be given support with any data they need to input using clear instructions. For example, if they are asked to set a password, they should be told in clear instructions what kind of data the password should include.
Compatible	A website should be compatible with as many devices that could be used to access it as possible. This includes desktops, laptops, tablets and mobile devices.

PRINCIPLES OF USABILITY

DESIGNING INTUITIVE NAVIGATION

One of the most important things that you need to consider when designing a website is how a user actually navigates their way around it. There are several different types of navigation systems that you can choose for a website:

- horizontal scroll menus
- vertical menus
- drop-down menus
- breadcrumb navigation
- button groups.

Horizontal scroll menus

KEY TERM

horizontal scroll menu: a navigation menu that can be scrolled that is displayed horizontally across the screen from left to right

A **horizontal scroll menu** is a navigation menu that is normally displayed across the top of the web page; it requires the user to scroll horizontally left and right to see all the options available in the menu.

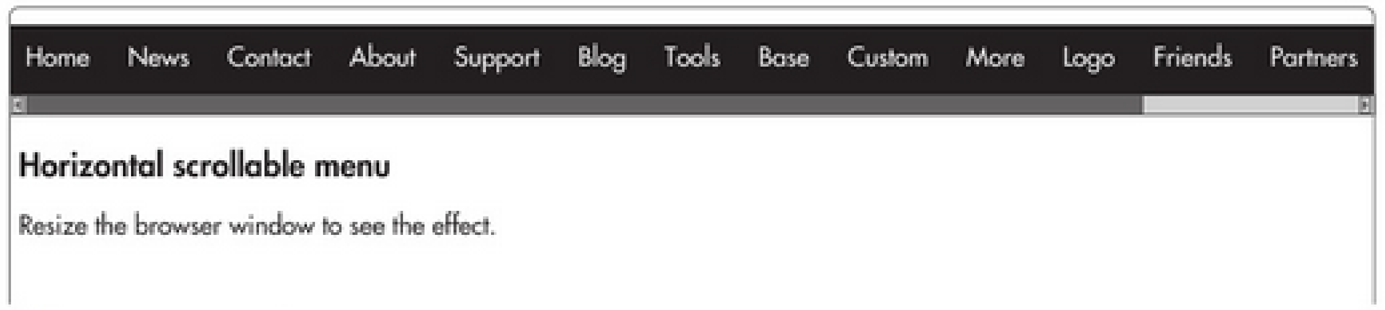
A horizontal scroll menu used to be a feature that designers were advised not to include in their website. However, they have now become more popular in modern websites. This is because mobile phones often have limited space on the screen, so it is necessary for a user to scroll to see some content. Designers were initially advised against it because it is more difficult for a user to scroll horizontally across the page than vertically up and down the page. However, on mobile devices a user will normally scroll across or down a screen using fingertip gestures, so it is as easy to scroll left to right as it is to scroll up and down.

A designer will normally need to give some indication to a user that the horizontal menu contains more options that they can scroll to. This can sometimes involve displaying a scroll bar beneath the menu, as shown in Figure 10.3.

KEY TERMS

drop-down menu: a navigation menu that displays further options for the user to select when the user hovers over or clicks an option in the menu

vertical menu: a navigation menu that is displayed from top to bottom on the screen, normally down the left side



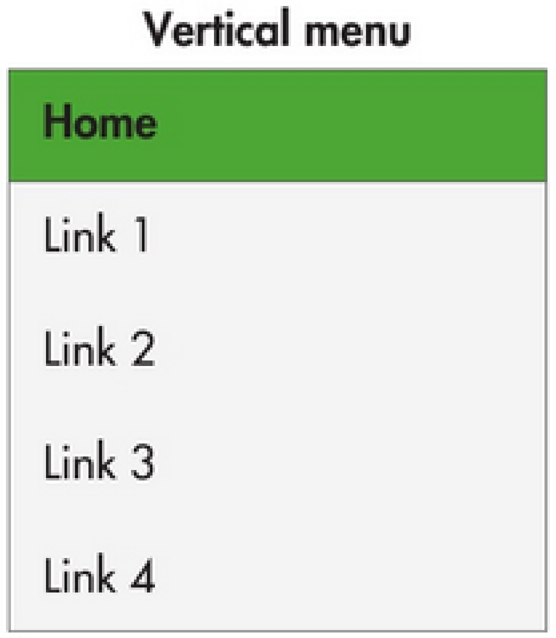
Horizontal scrollable menu

Resize the browser window to see the effect.

▲ Figure 10.3 A horizontal scroll menu

Vertical menus

A **vertical menu** is a navigation menu that normally appears down the side of the screen (see Figure 10.4). It is most commonly displayed down the left side of the screen. Vertical menus can be great for websites viewed on laptops and PCs as they generally have more screen space. A vertical menu is often a clearer way for a user to view the navigation links available on a website. However, vertical menus often take up a lot of screen space, so on a mobile device they can end up filling too much of the screen and blocking the user from immediately seeing any other content.

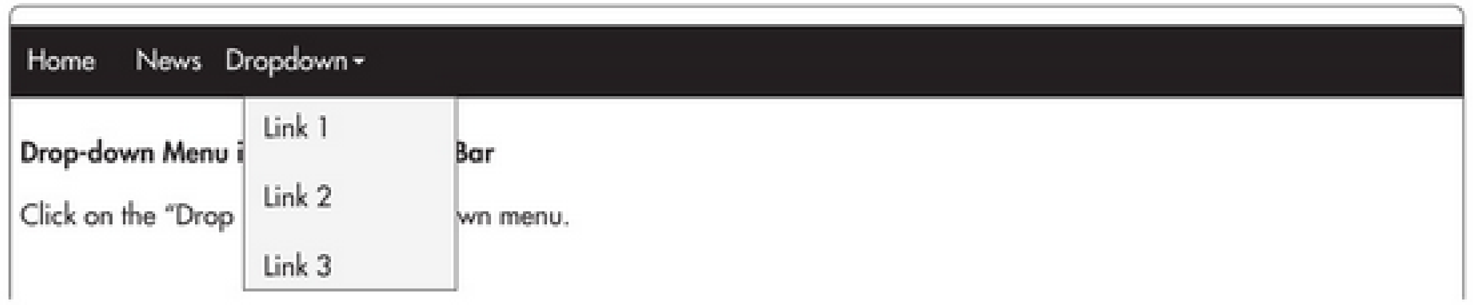


▲ Figure 10.4 A vertical menu

Drop-down menus

A **drop-down menu** is a navigation menu that displays further options in a box that opens when the user interacts with an option in the menu (see Figure 10.5). For example, a user may interact with an option in the menu by hovering over it. This will then cause a box to drop down from the menu option and display further options that relate to it.

Drop-down menus can be used in both horizontal and vertical menus.



Drop-down Menu i **Bar**

Click on the "Drop wn menu.

▲ Figure 10.5 A drop-down menu

Breadcrumb navigation

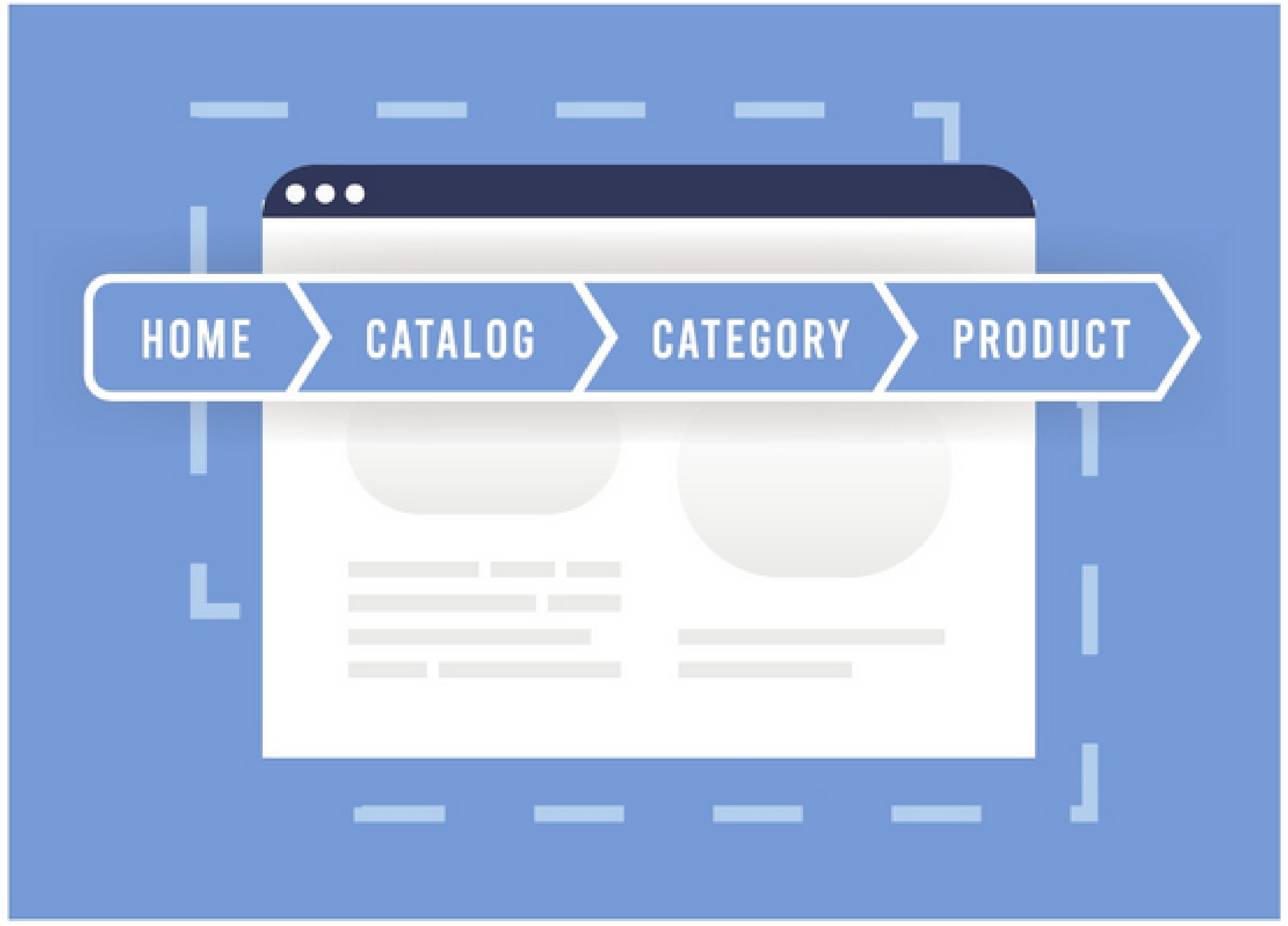
Have you ever heard the fairytale story of Hansel and Gretel? In the story they have to go into a forest away from their house. They leave themselves a trail of breadcrumbs so that they know

KEY TERM

breadcrumbs: a trail of links to show how the user has navigated to a certain web page on the website

where they have been and can find their way back. **Breadcrumbs** are used in a very similar way in website navigation (Figure 10.6).

The website will display a trail of links to show how the user has navigated to a certain web page on the website. They can then click on any of the links to find their way back to an earlier web page that they visited. For example, a user opens a shopping website at the home page. They then click the menu link for the catalog, choose a category and then click the link for a product. They would end up with breadcrumbs displayed on the page that look like this:



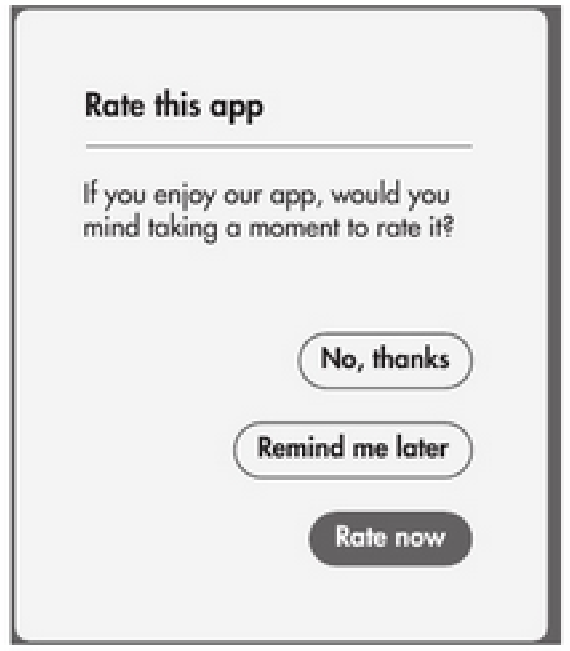
▶ Figure 10.6 Breadcrumb navigation

Button groups

A button is an icon on a web page that a user can click to navigate to another part of the website. Web pages often include several different buttons. Some examples of common buttons included in a website are:

- Cancel
- Clear
- Submit
- Rate now
- Remind me later
- Continue
- Share.

Designers will often group buttons together that have functionalities that relate to each other. Buttons are normally grouped either horizontally or vertically. For example, you might get a pop-up box when you use a website that asks you to rate the website (Figure 10.7).


Rate this app

If you enjoy our app, would you mind taking a moment to rate it?

▶ Figure 10.7 A pop-up box

A pop-up box like this has three buttons that are grouped together vertically. You would normally use button grouping to make it easier to find all the options that are available for a certain part of the web page. You would also often use them when creating web forms where users are required to fill out their personal details.

KEY TERM

cross-browser compatibility: a website that has the design and function to work on many different browsers and devices

TARGETING SPECIFIC DEVICES AND BROWSERS

Cross-browser compatibility

A very important issue to consider when designing and developing a website is that the website could be viewed on hundreds of different devices using several different web browsers. This means it is important for a website to have **cross-browser compatibility**. Therefore, you must create the website so that it will display as identically as possible across different web browsers and devices. This will involve making sure that the website can adapt to browsers that may use older methods of displaying web pages that do not support the latest features of web development. It will also require being sure that the website can adapt to run on lots of different types of hardware. You will want to make sure that no matter the device or browser, the user can see all the information that they need to, and that they can interact with the web page in all the ways that they need to.

Safari
Edge
Chrome
Firefox
Opera

▶ Cross-browser compatibility is important for a website

Functionality and user testing

To make sure that a website has cross-browser compatibility, you may need to carry out lots of functionality and usability testing. This will involve asking lots of users with different devices and different browsers to test the website. They will test the functionality of the website and make sure they can complete all the necessary actions that are needed. They will also make sure that the website is usable on each device and browser. This type of testing will involve looking at how the information on the website is displayed and interacting with functions such as navigation.

There are online tools (e.g. BitBar and Browsera) that a web developer can use to test whether their website is compatible for many different browsers. Some of these tools offer free testing, while others require a subscription payment. The developer enters the URL of the web page and the online tool shows how that web page would be displayed on many different browsers. Some will also show how the web page would be displayed on different versions of a browser.

Code validation

Once a website is created it should go through a process called **code validation**. The World Wide Web Consortium (W3C) sets standards for the world wide web. It has a set of formatting standards for a website that can be used as a code validation test. These formatting standards are designed to show web developers what best practice is for formatting a web page. If the code for your website follows the standards set by W3C, your website is likely to appear higher up in search engine results. One of the main aims of the standards is to make sure that websites are suitable and display correctly on all web browsers. They are a way of checking for cross-browser compatibility. The W3C has a tool available that can be used for code validation. There is both an HTML validator and a CSS validator.

KEY TERM

code validation: testing a website to see if it meets the required criteria

Browser development tools

Most modern web browsers have a selection of development tools that can be used when developing a website. These tools will inspect the code and make sure that it is suitable to run on the browser. They will also provide you with information, such as how long it took the website to load in the browser. Some web browsers also have tools that will allow a developer to make small changes to the website while open in the browser.

CHECKPOINT

1. What is the difference between a wireframe and a mock-up?
2. Explain what kind of things can be found in a style guide.
3. Identify and describe three of the WCAG areas.
4. Where are horizontal scroll menus mostly used?
5. What is breadcrumb navigation?
6. Define what is meant by cross-browser compatibility.
7. What is the World Wide Web Consortium?
8. What is code validation?

THINKING LIKE AN IT SPECIALIST

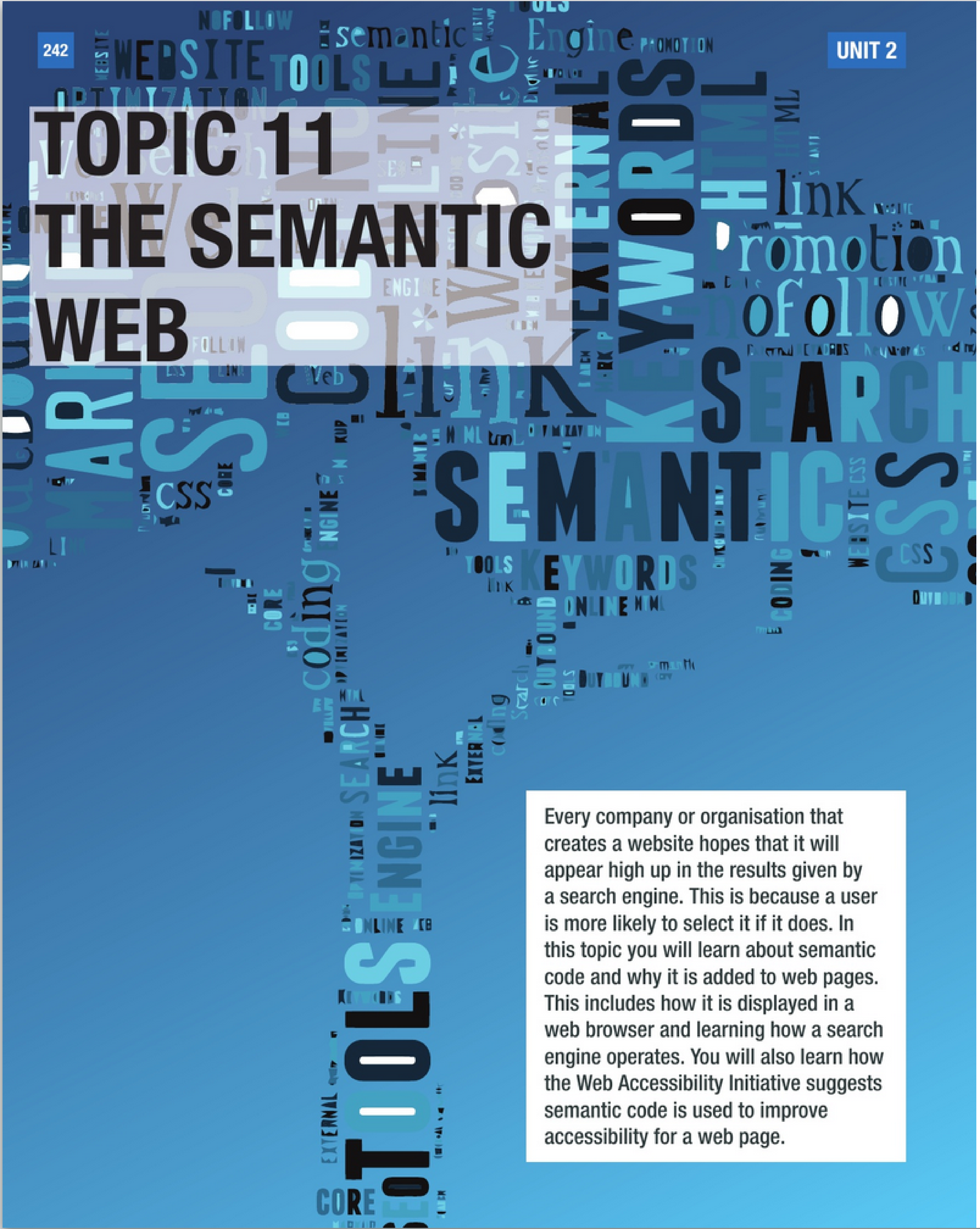
The World Wide Web Consortium is the organisation that sets the standards for the world wide web. The aim of their standards is mostly to make sure that all websites are accessible to as many people as possible. What kind of standards would you want to see from a website if you were part of the World Wide Web Consortium?

SKILLS

INTERPRETATION, CRITICAL THINKING

EXAM PRACTICE

1. Web designers normally use web-safe fonts when creating a web page.
   - (a) Give an example of a web-safe font. [1 mark]
   - (b) Explain what is meant by a web-safe font. [2 marks]
   - (c) Give one reason why a web designer uses web-safe fonts. [1 mark]
2. Wireframes and mock-ups can be used in the process of designing a website.
   - (a) State what is shown in a wireframe. [1 mark]
   - (b) Assess how useful a wireframe and a mock-up are when designing and developing a website. [6 marks]
3. Explain what is meant by a website that has cross-browser compatibility and why it is important. [4 marks]


SEMANTIC
SEARCH
KEYWORDS
HTML
link
promotion
nofollow
WEBSITE
TOOLS
semantic
Engine
OPTIMIZATION
EXTERNAL
CSS
coding
ONLINE
OUTBOUND
CORE

TOPIC 11 THE SEMANTIC WEB

Every company or organisation that creates a website hopes that it will appear high up in the results given by a search engine. This is because a user is more likely to select it if it does. In this topic you will learn about semantic code and why it is added to web pages. This includes how it is displayed in a web browser and learning how a search engine operates. You will also learn how the Web Accessibility Initiative suggests semantic code is used to improve accessibility for a web page.

TOPIC 11 THE SEMANTIC WEB

LEARNING OBJECTIVES

By the end of this topic you should be able to understand:

- what is meant by semantic markup on a web page
- how semantic markup is used by web browsers and search engines
- how non-semantic markup was used in older methods of web development
- how to add semantic markup to add structural meaning
- how to add semantic markup to add textual meaning
- how the Web Accessibility Initiative – Accessible Rich Internet Applications (WAI-ARIA) can be used to provide additional semantics for accessibility.

GETTING STARTED

Try using assisted technology on a web page. The simplest way to do this is to use a screen reader. Google has one available as an extension in its browser that you could use.

Close your eyes and listen to the audio given by the screen reader. How well do you feel you know what the web page looks like and the content it contains when the screen reader has finished describing it?

VOICE ASSISTANT
HI! HOW CAN I HELP YOU?

▲ A smartphone voice assistant offering help

KEY TERMS

HTML 5: the latest version of a markup language that is used to develop web pages. It describes the structure and presentation of the web page

HTML tags: key words that are included in web pages to define structure and presentation

semantic: the meaning of a word

SEMANTIC MARKUP

The word **semantic** relates to meaning. The semantics of language is the meaning of the words that are used in the language. In a similar way, semantic markup is the meaning of each element that is created in a markup language (e.g. HTML) in a web page.

Many **HTML tags** have semantic meaning. This means that they show some information about the type of content that is given between the opening tag and the closing tag. Browsers use the semantics of these tags to know what different content on the page is and how to treat it for display.

For example, if a paragraph tag (<p>) is used for each element of text that is given on the page, the browser would treat all text the same as it all has the same tag. However, if the tag <h1> is used to define an element that has a heading in it instead of a paragraph of text, the browser knows that this is the main heading for the paragraph of text. This becomes important when the browser has to render the HTML on different devices, as it is able to use the semantics of each element to know how to best place the elements to adapt the web page for different devices.

Therefore, one of the most important things to do when you are writing a web page is to use the correct tags for each element. This gives the browser the semantics of the content that is included in each element.

USING SEMANTIC MARKUP FOR STRUCTURE

Semantic markup can be used to tell the browser what different areas of the web page are. It gives the browser some guidance about what the structure should be for the web page. However, the kind of tags used to provide information like this only became available with the release of **HTML 5**. Prior to HTML 5, the two main tags that were used to structure a document were the <div> tag and the <span> tag.

The <div> tag defines a section of a document. Any content that is included between an opening and closing <div> tag is part of that section. The <div> tag is a block tag. This means it will stretch the full width of the available space.

The <span> tag is similar to the <div> tag; the main difference between the two is that the <span> tag is an inline tag. This means that it will only stretch the width needed to display the content included between the opening and closing tags.

These tags were the tags available in older styles of web development. Developers would use block tags like the <div> tag to define the sections of the web page. They would use inline tags such as the <span> tag to format certain areas within those sections.

SKILLS

CREATIVITY, INTERPRETATION

ACTIVITY 1

Try using the <div> and <span> tags in a new blank web page.

Use the <div> tag to create a banner section to be displayed across the top of the page. Add your name and find an image to use as a small logo in your banner section.

Use the <span> tab to create a navigation section to be displayed below your banner. Add three options to your navigation section.

View your web page in a browser and have a look at the difference between how the <div> tag renders and how the <span> tag renders.

Look back to Topic 7 for a reminder about how to use <div> tags.

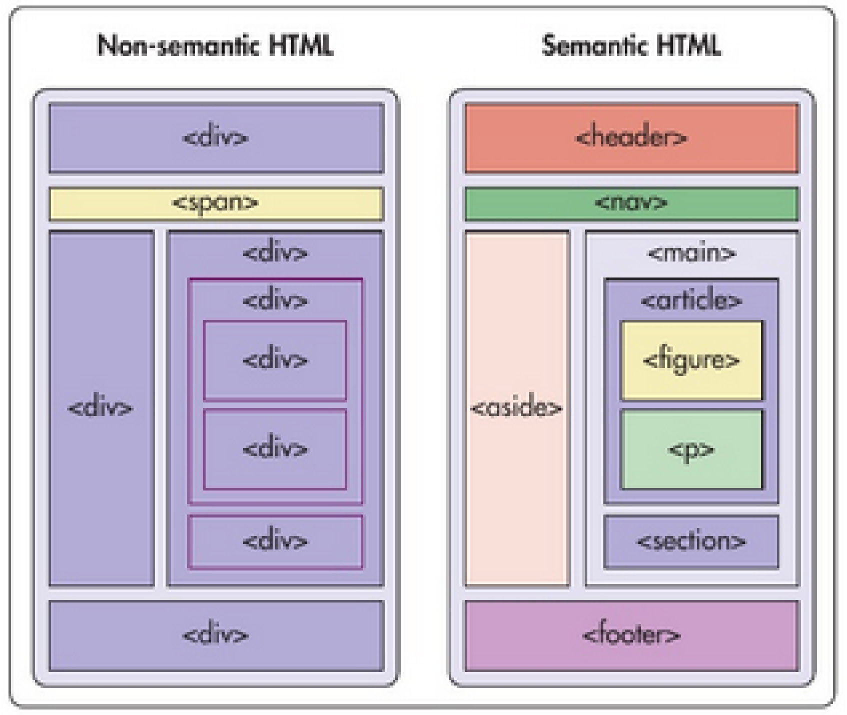
The <div> and <span> tags are non-semantic tags. They give the browser very little information about the content that is included between the opening and closing tags. The release of HTML 5 gave web developers a set of new semantic tags that can be used to provide the browser with more information about the structure of the web page.

Table 11.1 shows some of the different areas of a web page and a description of how semantic markup is used for each.

TABLE 11.1: CONTENT AREAS AND THEIR SEMANTIC MARKUP

Content area	Semantic markup
Headers and footers	A <header> tag is used to show where the header for the web page begins and ends. The type of content normally included in a header tag is a company's name and logo and sometimes navigation elements. A <footer> tag is used to show where the footer for the web page begins and ends. The type of content normally included in a footer tag is author information, contact information, copyright information and sometimes some additional navigation links.
Navigation	A <nav> tag is used to create a navigation section for the website. This will normally contain links that are used to navigate to the different sections and web pages within the website.
Main	A <main> tag is used to show where all the unique content for a single web page within the website begins and ends. Content that is duplicated on each web page in the website is not normally included in this tag. This tag replaced the <body> tag that was previously used for a similar purpose.
Sections	A <section> tag is used to define different sections of a web page. It is often used in long web pages that would benefit from being divided into sections. A section tag should be used for each individual section of a web page. A good way to think about where a section tag should be used is that if the content has a new heading then a section tag should be used to tell the browser this is a new section of content.
Articles	An <article> tag is used to display content that is a standalone and independent part of a web page. This type of tag is often used to display content such as blog or forum posts, or news stories. Using an article tag will create distinct sections for a web page.
Asides	An <aside> tag is used to display content that is related to the main content on the page but not directly part of it. It is often used to display related links, to define terms used in the main content, for pull quotes, or to add comments or a bibliography to the main content.

Figure 11.1 shows how a web page with and without semantic markup would look.



▶ **Figure 11.1** How a web page with non-semantic markup and one with semantic markup would be structured

SKILLS

CREATIVITY, PROBLEM SOLVING

ACTIVITY 2

Create a web page about your favourite place to visit using semantic markup. The structure for your web page should match the structure given in Figure 11.1. You will need to choose content to be displayed in each of the different sections. You do not need to create more than one page, but you should have multiple options available in your navigation bar to represent other web pages.

If you are struggling to think of what kind of content to add to a section, you could talk to a friend in your class about what their plan is.

Once you have created your web page, make sure you compare it to Figure 11.1 to see if you have managed to create it using the same structure.

KEY TERMS

search engines: software tools that find web pages that match search criteria given by a user

search engine crawlers: search engine bots that connect to websites on the internet to download and index their content

HOW DO SEARCH ENGINES USE SEMANTIC CODE?

The use of semantic code in websites is important for **search engines**. The internet is not a very organised place. It is not like a library where all the information is ordered to make it easy for a person to find what they need. The internet has information all over the place that has very little organisation to it. This means that search engines have a challenging role to find the information requested in a search.

▶ The internet has very little organisation. It is not like a library where books are organised and ordered

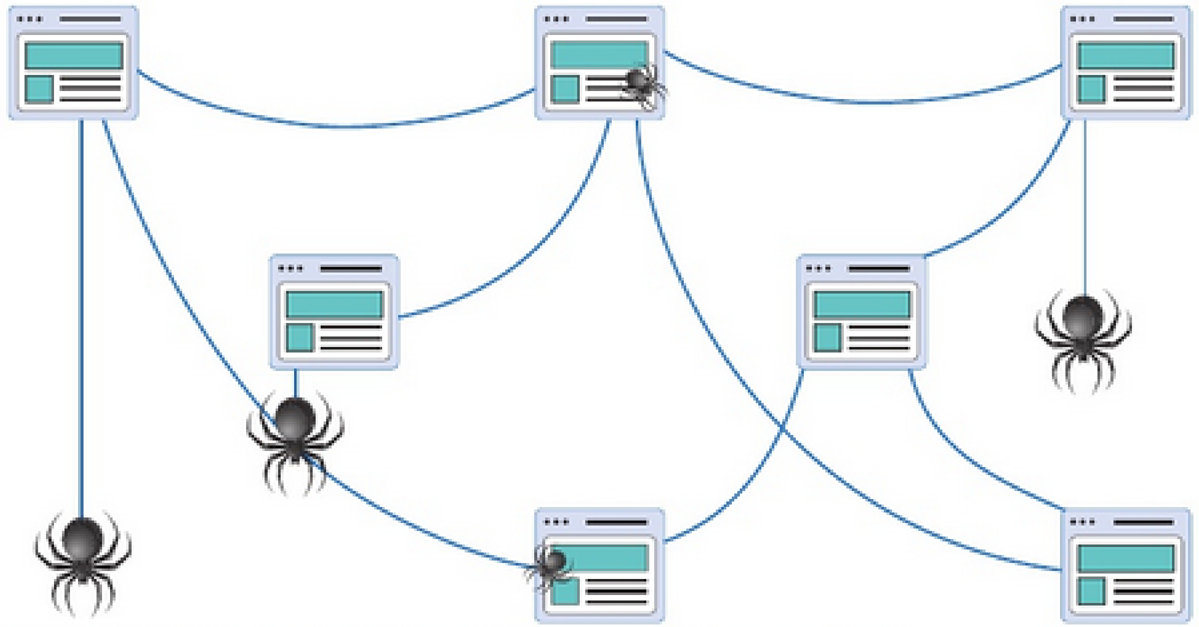
Search engines regularly send out lots of little bots called **search engine crawlers** (sometimes known as spiders). They are sent off to search the internet for any new information and web pages (see Figure 11.2). These crawlers are algorithms that search through large amounts of information. Any new information found is organised and added to the search engine's database. This process is called indexing.

When a user types their search criteria into a search engine, complex algorithms are used to search the database. If a web page has used semantic markup, this makes it easier for the algorithm to quickly search your website to see if it matches the criteria. The results of the search are organised into an order by the search engine. The order depends on what the search engine algorithms find to be the best results to match the search criteria. This process is called ranking. Websites that use semantic markup have a much better chance of appearing higher up in the search results. This is because the use of semantic markup makes the website much more search friendly. If a website is not search friendly, the algorithm may not even look through it.

KEY TERM

search engine optimisation (SEO): the process of improving the visibility of a website in search engine results

Adding semantic markup to your website for this reason is part of a process called **search engine optimisation (SEO)**. This is where an organisation will try to maximise the number of visitors it gets to their website by making sure that it appears high up in the search results for a. This is important because most people will often choose to look only at the first few results for the information they want.



▲ Figure 11.2 Crawlers regularly search the internet to find new web pages

USING SEMANTIC MARKUP FOR TEXTUAL MEANING

Semantic markup can also be used to add meaning to the text that will be included between the opening and closing tags. It can tell the browser what kind of text it is and the importance the text may have. Table 11.2 shows some of the different types of text that can be displayed on the web page and how semantic markup is used for this.

TABLE 11.2: TYPES OF TEXT AND THEIR SEMANTIC MARKUP

Type of text	Semantic markup
Important text	The most important headings in the web page will be given the tag <h1> and any further subheadings on the page may be given a tag between <h2> and <h6> depending on how important they are. It is also possible to make important text stand out on the page by using the tag. This will make any text between the opening and closing tags appear bolder than other text. For example: <p>My favourite food is pizza .</p> would display as: My favourite food is **pizza**.
Emphasised text	The tag can be used to emphasise any text on a page. It makes any text between the opening and closing tags display in italics. For example: They really loved pizza! would display as: They really *loved* pizza!

(Continued)

TABLE 11.2: TYPES OF TEXT AND THEIR SEMANTIC MARKUP (*Continued*)

Type of text	Semantic markup
Quotations	Two tags can be used for quotations on a page. The choice of which to use will depend on how the designer and developer want the quote to be displayed. If the quote is embedded within a paragraph of text then the <q> tag will most likely be used. This tag adds quotation marks to any text between the opening and closing tags. For example: She really loved pizza so she said to her friend, <q>Let's go out for pizza!</q> would display as: She really loved pizza so she said to her friend, "Let's go out for pizza!" If the quote is larger, the designer and developer may want it to appear more distinctive from other text on the page. To do this the <blockquote> tag can be used. This tag indents any text between the opening and closing tags. For example: A pizza lover's point of view: <blockquote> I just think that pizza is the most awesome food. It makes me happy. </blockquote> Maybe everyone should try it! would display as: A pizza lover's point of view: I just think that pizza is the most awesome food. It makes me happy. Maybe everyone should try it!
Abbreviations and acronyms	If text includes an abbreviation or an acronym, the designer and developer may want to provide the user with the full expansion of the acronym within the text, or they may want to have a function where the user can hover over the acronym to find out the full expansion. For this purpose they can use the <abbr> tag. An acronym that is included between the opening and closing tags can be given a title text that will display when the user hovers over the acronym. For example: <abbr title="Cascading Style Sheet">CSS</abbr> is used in web development. would display as: CSS is used in web development. When the user hovers over CSS a box will display 'Cascading Style Sheet'. The <abbr> tag is normally the tag used for this purpose; however you can also use the <acronym> tag. This is an older style of tag but it works in the same way as the <abbr> tag.

(*Continued*)

TABLE 11.2: TYPES OF TEXT AND THEIR SEMANTIC MARKUP (*Continued*)

Type of text	Semantic markup
Citations and definitions	The <cite> tag is used to give information about the author of a text. It is often paired with quotations so that the author of the quotation can be given. Any text between the opening and closing tags is normally displayed in italics. For example: To be, or not to be, that is the question. ~ <cite> William Shakespeare </cite>, from Hamlet. would display as: To be, or not to be, that is the question. ~ *William Shakespeare*, from Hamlet.
Addresses	The <address> tag is used to identify contact information for an individual or a company. Any text between the opening and closing tags is normally displayed in italics. It will also add a line break before and after the address block. For example: Contact us at: <address> 123 First Street First town 11201 </address> We look forward to hearing from you! would display as: Contact us at: *123 First Street* *First town* *11201* We look forward to hearing from you!
Marks	The <mark> tag can be used to highlight text. Any text between the opening and closing tags is highlighted in yellow. For example: I <mark> love </mark> pizza! would display as: I love pizza!

(*Continued*)

TABLE 11.2: TYPES OF TEXT AND THEIR SEMANTIC MARKUP (*Continued*)

Type of text	Semantic markup
Captions	The <caption> tag is used to add a title to a table. The title is normally centrally aligned above the table. The caption tag must be used straight after the table tag. For example:

```
<table>
 <caption>Daily temperature</caption>
 <tr>
  <th>Highest temp</th>
  <th>Lowest temp</th>
 </tr>
 <tr>
  <td>40 degrees Celsius</td>
  <td>28 degrees Celsius</td>
 </tr>
</table>
```

would display as

Daily temperature	
Highest temp	Lowest temp
40 degrees Celsius	28 degrees Celsius

Figures	The <figure> tag is used to identify a self-contained item. It is normally used to contain items such as images, illustrations and diagrams. It is often combined with the fig caption tag to add a caption to the item. For example:

```
<figure>
 <img src="pizza.jpg" alt="pizza" style="width:100%">
 <figcaption>This is pizza!</figcaption>
</figure>
```

would display as:

▲ This is pizza!

SKILLS

CRITICAL THINKING, CREATIVITY

ACTIVITY 3

Open the web page you created in Activity 2.

Try applying each of the semantic markup tags from Table 11.2. You may be able to use content that you have already written. If you don't have quotes in your web page, you could add a quote from yourself as a review of your favourite place.

WEB ACCESSIBILITY INITIATIVE – ACCESSIBLE RICH INTERNET APPLICATIONS (WAI-ARIA)

The WAI-ARIA is a technical specification that was released by the World Wide Web Consortium. It provides information about how a web developer can increase the accessibility of a website. You learned about some of the guidance from this in Topic 10.

Every website should be accessible to a person regardless of any disabilities. WAI-ARIA identifies some additional semantics that can be added to HTML tags to make websites more accessible. It is mainly used by assistive technologies such as screen readers. This is done using four main additions:

- roles
- states and properties
- live regions
- enhanced keyboard navigation.

Roles

Roles are information added to an HTML tag to show what role the element has. For example, a 2D icon on the screen may be intended as a button. However, when a screen reader is used by a person who is visually impaired, the screen reader may not recognise that the button has the role of being a button. To improve this, a role instruction can be added to the HTML tag to tell software such as screen readers that there is a button on the screen. This would mean adding the text role ="button" to the tag when creating the button.

There are many other parts of a web page that can be given role semantics, including:

- menus
- toolbars
- images
- lists
- forms
- links.

States and properties

States and properties can be paired with roles to show what the current state of an element is. For example, a website may have a check box that the user can check or leave unchecked to say whether they would like to join the mailing list. A screen reader may not be able to tell whether the box is currently checked or unchecked. Adding a further instruction to the HTML tag can tell the software whether the box is automatically checked or unchecked. This would mean adding the state aria-checked="false" to the tag when creating the check box.

You can add a state or property to the same elements that you can add a role to.

Live regions

Sometimes content will change on a page without the page being reloaded. For example, if a user adds an item to their shopping cart, the page may not refresh but a user who can see the screen will probably be able to tell that an item has been added to their cart. They may see a number 1 at the side of their shopping cart, or they may be able to see a small visual of the item added to their cart. However, a visually impaired user who is using a screen reader will not know whether the item is added to the cart. This is because the screen does not refresh, so the screen reader will continue to describe the remainder of the screen.

A live region can be identified on the website so that assistive technology will know to alert the user that something has changed. A live region is created by adding the aria-live attribute to the relevant HTML tag that has the content that may change. The instruction that is normally added is aria-live = "polite". This option means that when the user stops interacting with the screen it will inform them that the content in a live region has changed. The instruction aria-live = "assertive" can also be used. This will interrupt any announcement the screen reader is currently making to tell the user that content has been updated. This option is normally only used for time-critical updates, otherwise it could become quite irritating to the user if the screen reader is interrupted too much.

Enhanced keyboard navigation

The fourth semantic that should be included in a website is that it must be keyboard accessible. This can be achieved by adding aria-keyshortcuts to a tag, with the required keys then given for the shortcut. For example, aria-keyshortcuts = "Control+Alt+Delete".

SKILLS
CRITICAL THINKING

ACTIVITY 4

Use the internet to find out the difference between using HTML5 semantic markup and aria-controls to improve accessibility in a website.

CHECKPOINT

1. What is meant by semantic markup?
2. What are the <div> and <span> tags used for?
3. Give two HTML5 tags that can be used to add semantics to structure.
4. Which HTML5 tag has mainly replaced the <body> tag?
5. Why is semantic markup added for textual meaning?
6. What is search engine optimisation?
7. How does semantic markup assist search engine optimisation?
8. What are roles?

THINKING LIKE AN IT SPECIALIST

Semantic markup was introduced to make websites more accessible. It also helped to improve search engine optimisation. What sort of things would be important if you were tasked with search engine optimisation for your company's website?

EXAM TIP

It would help your understanding to use several different web browsers. Try opening the same web page in each browser and see what the differences are in how the web page displays. Try doing this with three different web pages and look at whether there is a pattern in how certain web browsers display certain things on the web page.

SKILLS
CRITICAL THINKING, INTERPRETATION

EXAM PRACTICE

1. Explain what is meant by semantic markup in a web page. [2 marks]
2. Identify which HTML tag is a non-semantic tag. [1 mark]
   A <header>
   B <main>
   C <section>
   D <div>
3. Explain why it would be better to use the <nav> tag to creative a navigation bar instead of the <span> tag. [2 marks]
4. Explain what is meant by live regions in the Web Accessibility Initiative – Accessible Rich Internet Applications (WAI-ARIA) technical specification. [2 marks]

GLOSSARY

absolute size a size that does not change and will remain constant

access rights restrictions that are placed on who can see and edit data

active digital footprint data you share online deliberately

ad hoc network a temporary network that is set up to share resources

analogue data that is continuous in nature and is the data that is processed by humans

anti-malware software that scans a device and removes or quarantines malware

application software that is designed to perform a specific task

application programming interface (API) an interface between a programming language and another application that defines how they will communicate

application software a type of software that provides services needed by the user

array a type of data structure, which, unlike a variable that only stores one value, can store multiple values

artificial intelligence (AI) an IT system that has the ability to learn and adapt. It is designed to mimic the intelligent behaviour of a human

attributes define a property of an element

authentication a collection of security methods that are used to verify the identity of a user when they are trying to access a device or an account

automatic software update a setting that can be applied to software updates to install them without a user's intervention

availability a measure of the amount of time a network is available for use

bandwidth the maximum amount of data that can be transferred from one device to another in a set amount of time

base station a large tower that is used to transmit data across a cellular network

basic input/output system (BIOS) a set of instructions that are part of the firmware that control the input and output operations

binary a number system that uses values 0 and 1

biological data data that belongs to a human, for example a fingerprint, an iris, facial features and their voice

biometric device a type of security device that uses biological data as a form of password

block elements HTML elements that start on a new line and occupy the full width of the browser window

Bluetooth® a type of wireless data transmission that uses radio waves to transmit data between two devices that are in close range (within 10 m)

Bluetooth chip a piece of hardware in a device that allows it to have a Bluetooth connection

blur when focus moves away from a particular element then a blur event occurs

bootloader a program that is part of the firmware; it is responsible for loading the operating system when a computer is turned on

breadcrumbs a trail of links to show how the user has navigated to a certain web page on the website

bridge network hardware that is used to connect one LAN to another LAN

cache a type of storage that stores the most frequently requested web pages and data

cellular network a type of network made up of cells that each have a base station that is used to transmit signals used by mobile phones

checkbox a form control where any of the options can be selected

civil liberties a person's right to basic freedoms as long as they respect other people's rights

client a computer in a client–server network that is not a point of control in a network; it accesses data and services on a server

client–server network a type of network that has central servers working as a point of control in the network and client computers that connect to them

client-side scripting processing for a web page that takes place on the client's computer

cloud computing cloud storage and other services such as software applications that are available from a cloud computing organisation

cloud storage servers that provide data storage that you can access from a remote location using the internet

cloud technology technology that includes servers that provide data storage that you can access from a remote location using the internet

coaxial cable a copper cable that has a single central wire with insulating layers; it can be used to transmit data across a network

code validation testing a website to see if it meets the required criteria

colour theory the understanding of how colours work together and how they compare and contrast

combinator defines the relationship between selectors when using CSS

compatibility issues problems that can arise when an update to software causes the software to not function properly

composite key two fields in a table that are combined to create a unique identifier for the table

connectivity the ability to connect devices to other devices

content models grouping of elements that have a particular purpose or behave in a similar way

context the conditions around which something exists

copyright legal form of protection that can be applied to intellectual property; it gives the owner control over what happens with the property and who can use it

Creative Commons an organisation that provides different types of licences for software distribution

cross-browser compatibility a website that has the design and function to work on many different browsers and devices

data text, values, symbols, images and sound in their raw form

data collisions when two devices on a network send data simultaneously, the data packets can collide

data extraction the process of collecting data from various sources and processing it to retrieve meaningful information

dataflow diagram a diagram that shows how data flows through different parts of a system

data mining the process of collecting and analysing large databases to look for patterns and trends

data packet the unit of data that contains the information to be transmitted across a network

data type a way of grouping data; examples include text, number, currency, date, time, and dates

database a collection of related data that is related

decomposition breaking down a problem, design or system into smaller parts

denary a number system that uses values 0 to 9

design a plan of what the requirements are for a system

design brief the place where the design for a system is noted

device an electronic piece of technology

digital data that is discrete in nature and is the data that is processed by computers

digital divide the divide between people who have access to and can use technology and the internet, and those that do not

digital footprint a record of all of a person's activity online

disabilities conditions of the body or mind that make it more difficult for a person to carry out certain actions

Document Object Model (DOM) an API that gives JavaScript access to HTML

domain name system a type of server or service that stores a database of domain names and their matching IP addresses

domain name a part of a URL that has the name of the web page; for example, the domain name for the URL https://www.pearson.com is pearson.com

driver a type of systems software that allows devices to communicate with the operating system

drop-down menu a navigation menu that displays further options for the user to select when the user hovers over or clicks an option in the menu

dynamic host configuration protocol (DHCP) a protocol that is used to assign an IP address to a device when it is connected to a network

dynamic web page a web page where the content is generated every time the page is requested, so it may change and display differently for each user

electromagnetic waves a form of radiation that travels through the air

element a component of an HTML document

embedded system a type of system that has a dedicated purpose

encrypted data that has been scrambled so it is meaningless to anyone who does not have the key to unscramble it

encryption a method of security that scrambles data to make it meaningless

energy the amount of electricity used by a device

entity another name for a single table in a database

environment the surroundings or conditions in which a person, animal or plant lives or operates

error rate a measure of the number of errors that occur in a network

ethernet a protocol used to connect devices in a LAN using a wired connection

evaluate to assess and judge how suitable a system is and how closely it matches the design

expansion capacity the ability to add parts to a device, for example adding more storage space to it

external style sheet CSS styles defined in a separate document to the HTML

feature a distinctive part of a device, like its size and colour

fibre optic cable a cable that is made up of fine strands of glass or plastic

field an individual piece of data about an item that is stored in the database; it is a column in the database

firewall a type of security setting that can be used to examine data incoming and outgoing from a computer or network

firmware the basic instructions that allow a system to function

fit for purpose an evaluation to see whether a system meets all the original requirements from the design

flow the path a designer creates through a website for a user to follow

flowchart a diagram that shows the sequence of process and decisions that occur in a system

focus when an element receives focus it is ready to receive input from the keyboard or mouse

foreign key a primary key that appears in another table in the database to create a link between the two tables

forms a method of user input

format the way in which data is arranged

free software software that allows the user to study, modify and share the source code; a fee can be charged for the software

function a way in which a device can be used

gateway network hardware that acts as a translator between the two networks, converting data from one protocol to another to allow the networks to communicate with each other

general-purpose computer a computer that is used for a large range of different tasks

global positioning system a satellite-based navigation system

ground station a base on Earth that receives and transmits radio waves to a satellite

habitat the natural home of an animal or plant

hack to gain unauthorised access to data

hardware the physical parts of a device that you can see and touch

hexadecimal a numbering system using base 16 (rather than the base 10 system we commonly use). It is used in CSS for defining colours, with two hexadecimal digits used to represent the red, green and blue in a colour

horizontal scroll menu a navigation menu that can be scrolled that is displayed horizontally across the screen from left to right

hotspot what a mobile phone becomes when it allows devices to be tethered to it to use its internet connection

HSL stands for Hue, Saturation and Lightness and is another way of defining colours in CSS

HTML 5 the latest version of a markup language that is used to develop web pages. It describes the structure and presentation of the web page

HTML tags key words that are included in web pages to define structure and presentation

hub network hardware that devices connect to so that they can send data to each other in a network

hypertext a type of text displayed on a screen that contains links to other text

hypertext markup language (HTML) a type of language used to build web pages

image sprite an image made up of several images combined in the same file

inclusion the provision of equal access and opportunities to everyone

information data that has been processed by giving it context and meaning

infrared light electromagnetic radiation with wavelengths longer than visible light but shorter than radio waves; it can transmit signals only over very short distances

inline elements HTML elements that fit alongside the existing content (usually text) and don't need to start on a new line

inline style CSS styling added to the HTML element tag using the Style attribute

integrated circuit (IC) a set of small electronics circuits that are built onto a single electronic chip

intellectual property property that is the result of creative thinking and actions

internal style sheet CSS styles added to the individual HTML document in the head section

internet a network of computer systems and all the cables and other hardware that is used to connect them

internet of things (IoT) the everyday home devices (e.g. a fridge, washing machine, toaster etc.) and sensors (e.g. car park space monitors, temperature sensors) that are connected to the internet

interrupt a type of signal that is sent to the processor in a computer to gets its attention

IP address a unique address that is given to a device when it connects to a network

IT system all the hardware and software that are combined to create a whole system

jitter a measure of the disruption that occurs when data is transmitted across a network

landfill a place where non-recyclable waste materials are disposed of and buried in the ground

latency the delay that can be present in the transmission of data across the network

local area network (LAN) a type of network that is normally located within a single building or site

location awareness devices that are able to determine their own location

machine learning a type of artificial intelligence that uses data and algorithms to learn and adapt its processes

magnetic storage a type of storage device that uses electromagnets to store data

malware a type of malicious software that is used to damage data or a device

many-to-many relationship a link where many records in one table can be linked to many records in another table; this relationship normally has a middle table that creates two one-to-many relationships between the tables instead

media access control (MAC) address a unique address that is given to the NIC in each device at the point of manufacture

metadata information about data; in html it provides information such as the character set used and the purpose of the web page

microprocessor a small version of a processor where all the data is processed using a single or limited number of integrated circuits

microwaves electromagnetic radiation that can be used to create a wireless connection in a network; they operate only over short distances

mock-up a realistic representation of what a web page will look like

modem network hardware that converts between analogue and digital signals

monetise earn money from

multitasking when multiple tasks are executed by a computer at the same time

network interface card (NIC) a piece of hardware that is built into a device; it allows it to have a wired or wireless connection

non-volatile storage computer storage that retains its data when the power is removed

one-to-many relationship a link where one record in a table is linked to many other records in another table

one-to-one relationship a link where one record in a table is related to only one other record in another table

online community a place that exists online where people interact with each other by having discussions and sharing resources

online working environment a place where people share resources and communicate with each other about their work, using the internet

opacity refers to how transparent an element is

open-source software software that allows the user to study, modify and share the source code; a fee is not charged for the software

operating system a type of systems software that manages the hardware and software in a computer

OSI model a theoretical model that is the basis of many protocols

packet loss a measure of the number of packets that were transmitted from the sending device but did not arrive at the receiving device

pairing the name given to the connection made between two devices that are using a Bluetooth connection

passive digital footprint data that is collected about your activity online that you are not aware of

paywall a restriction placed on content that requires users to pay a fee to view the content

peer a single computer in a peer-to-peer network

peer-to-peer network a type of network that does not have a server as a point of control; all devices on a peer-to-peer network are equal

performance how well a device performs its intended purpose

peripheral a piece of hardware that is attached to a computer such as a keyboard, mouse, printer and headset

phishing a scam where a legitimate-looking email is sent to a person to try and obtain their personal data

physical controls physical security measures that can be used to keep data secure, such as CCTV and locks on doors

pollution the existence of a substance in the environment that has harmful effects on the environment and the things that live in it

port a connection point in a device that a cable is plugged into

portability how easy it is to carry a device around from one place to another

powerline a device that plugs into a power socket and transmits data over the electrical wiring to another device on the same circuit

primary key a field in a database that is used as a unique identifier for a record; this is because the data stored in the field is unique for each record

privacy having a place where you can go that your actions are not seen and monitored, and therefore cannot be judged by others

property assigns a particular style to an element

proprietary software software that is not distributed with the source code

protocol a set of rules that govern the transfer of data

prototype a working digital model of what a website will look like

qualitative data data that cannot be easily measured, such as written descriptions

quantitative data data that can be measured, such as a set of numeric values

query a search that is created to find specific data in a database

quick response (QR) code a type of barcode that can store data such as a link to a website

radio buttons a form control where only one of the group of options can be selected

radio waves electromagnetic radiation that can used to create a wireless connection in a network; they operate over longer distances than microwaves

random access memory (RAM) a type of volatile memory in a computer that stores currently running data and programs

read only memory (ROM) a type of non-volatile memory in a computer that is used to store start-up instructions for the computer

record a collection of all the fields about one item in the database; it is a row in the database

regular expression (regex) a pattern that describes the format an input should take

relational database a database that has multiple tables that are linked together using certain fields as keys

relationship a link that is created between two tables in a database

relative size a size that is relative to another element such as a parent element, and will change if the size of that element also changes

repeater a network device that is used to boost the strength of an electrical signal

repetition a programming construction where a block of code is repeated a certain number of times or until some condition is met

resources the materials that we have available to us

responsive design website design that responds to different screen sizes and orientations

RGB stands for Red Green and Blue, which are the primary colours, and which can be mixed together to make any colour

router network hardware that reads IP addresses to send data to the correct network or device

routing table a database that stores all the known routes across a network to different devices

satellite a device that orbits the Earth and uses radio waves to transmit data

scalability how well the capacity of a network can be increased or decreased to respond to different amounts of data transfer

search engines software tools that find web pages that match search criteria given by a user

search engine crawlers search engine bots that connect to websites on the internet to download and index their content

search engine optimisation (SEO) the process of improving the visibility of a website in search engine results

security settings ways that a user can control who can see or access data that they store in an account

selection a programming construction where a choice between two blocks of code is made using a condition

selector used to select which element the style will apply to

semantic the meaning of a word

semantic elements HTML elements that provide a definition of the purpose of the page content that the elements contain

sensor a type of input device that captures data from its immediate surroundings

server a central device in a network that stores data and can also provide services; it accepts and responds to requests that are made for different types of data

server-side scripting processing for a web page that takes place on the web server

SIP a protocol that is used as part of VOIP

smart cities cities that use technology, AI and machine learning to collect data, and monitor and adapt processes in the cities to make them more efficient

smart homes homes containing devices that can be remotely controlled by another device, such as a smartphone

software the instructions that tell a computer what to do

software patch a set of changes used to fix errors or improve functionality in software

software upgrade a whole new version of existing software that contains new features and functionality

solid-state storage a type of storage device that stores data on silicon chips

speed the speed at which the data is transferred from one device to another

static web page a web page in which the content is the same for every user that views it (unless changed by the author)

storage a piece of hardware that is used to store data

storage media the type of storage that can be used for a device; for example, the device can have a memory card slot or a USB port that allows a USB drive to be connected

strong password a password that has a mixture of letters, numbers and characters

structured data data that has a specific format

structured query language (SQL) the programming language that is used to create a query

style guide a document that defines all the style formatting elements for a website

sub-system a smaller individual section of a larger system

supercomputer a computer with a very high level of performance

switch network hardware that reads MAC addresses to send data to the correct device

systems software a type of software that provides services needed by the computer

table a collection of data that is related and structured in rows and columns

tag an item of markup that indicates the beginning and end of an HTML element

TCP/IP model a protocol that is used to transmit data across the internet

technological convergence when two or more devices are combined to create a single device, product or service

tethering the process of connecting another device to a mobile phone to make use of the phone's internet connection

throughput the amount of data that actually passes through the network from one device to another in a set amount of time

touchscreen a type of input device that allows a user to touch a screen to input data and use the screen to see the data

transform change an element's position and/or orientation

transistor a very small component in a computer that controls the flow of electricity

transition a gradual change form one property value to another

transmission media the types of cables or signals that are used to transmit data across a network

twisted pair cable a copper cable that has two twisted wires; it can be used to transmit data across a network

uniform resource locator (URL) a text-based address for a website

unstructured data data that has no specific format

user interface the part of a device that a person uses to interact with the device

validation involves checking that user input meets the required format or pattern

value properties are set to a particular value

variable a memory area used by a program to store data while the program is running

vertical menu a navigation menu that is displayed from top to bottom on the screen, normally down the left side

virtual private network a secure connection between devices that uses encryption to send data

virtually interacting using the internet instead of interacting in a physical place

visual hierarchy using different formatting on items to make them stand out more than others

VOIP a protocol that is used to send data when one person is making a voice call to another person using the internet

volatile storage computer storage that loses its data when the power is removed

web authoring a process of writing the content that is needed to build a web page

web browser software that is used to access web pages

Web Content Accessibility Guidelines (WCAG) guidelines that set standards for what should and should not be included in websites to make them more accessible

web server hardware that is used to store web pages and respond to requests for those web pages

wide area network (WAN) a type of network that is normally spread across several buildings or sites; the largest example is the internet

Wi-Fi a form of wireless access to the internet

wired connection a method of connecting devices using cables

wireframe a 2D blueprint of the layout of a web page

wireless access access to the internet that does not require any cables

wireless access point (WAP) network hardware that transmits and receives radio wave signals

wireless connection a method of connecting devices using radio waves

wireless network interface card (WNIC) a piece of hardware in a device that allows it to connect to a wireless network

world wide web the collection of web pages and websites that are viewed using the internet

World Wide Web Consortium (W3C) an organisation that sets standards for the world wide web

ZigBee® a type of wireless data transmission that uses radio waves to transmit data between two devices; it is one of the main methods of connection used in the Internet of Things

INDEX

Pages in italics are figures; with 't' are tables.

A

access, moral/ethical issues 122t
access key 133
access rights 59
accessibility 123t, 137, 231, 231-6t, 251-2
ad hoc networks 44, 45, *45*
adaptability 232t
addressing in a network 56–7, 57t
airport check-in system 81t
amending records from a database 106–7
animations, CSS 188–92, *189–90*
anti-malware 59
application software 24
applications, and performance 5, *5*
artificial intelligence (AI) 95
attributes
  CSS 169–71
  HTML 130
audio and video in web pages 152–3, 153t
authentication 58–9, 58t
automatic software updates 28t
availability, network 48t

B

bandwidth 18t, 47t
basic input/output system (BIOS) 17
battery power 14t
Berne Convention 119
binary numbers 19–22
biometrics 13t, 58t, 59, 74t, *75*
Bluetooth 50
bootloader 17
box model (CSS) 165–9, *166–8*, 179–80, *180*, 183–4
breadcrumb navigation 237–8, *238*
bridges 35t
browser development tools 240
button groups 238–9, *239*

C

capacity of a device 18t
CAPTCHA 231t
cellular networks 52–3, *52*
character sets 131
civil liberties, moral/ethical issues 122t
class attribute 133
client-server networks 42–3, 42t, *43*
client-side scripting 67, 67t
cloud, the 76–7, 76–7t, 118
coaxial cables 38, *38*
code validation 240
colours 171–5, *174*, 229–30t
combinators 162–4
communication protocols 49–55, *52*, 54t, 55t
connectivity 8–9, *9*, 15t
content models 135, 135–6t
copyright 119
counting records 109–10
Creative Commons 27t
crime, computer-related 120–1
cross-browser compatibility 239
CSS (Cascading Style Sheets)
  animations 188–92, 189–90
  positioning content 181–8, *181*, *185–6*, *188*
  styling web pages 171–81, *174*, *180*
  writing 159–71, *161–2*, *165–8*

D

data
  file size 22–4, 22t
  and information 94
  selling customer 72t
  sharing personal 73–5, 74–5t, *75*
  structured/unstructured 94–103, *98*, 98t, 99t, 100t, *103*
data attribute 133
data extraction 97–8, *98*
data protection laws 118–19
databases
  creating relationships between entities 101–3
  structured data 98–100, 98t, 99t, 100t
  structured query language (SQL) 103–12, 111t
dataflow 82–8, 83t, *84*, 85t, *86–8*
decomposition of a system 82
deleting records from a database 106–7
denary numbers 22
description, HTML 131
design principles, web pages 222–30, *224–5*, 228–300t
designing and evaluating IT systems 88–91
digital footprints 69–70, *70*
distinguishability 233t
doctypes 130
Document Object Model (DOM) 195–6
domain names/domain name system (DNS) 79
drop-down menus 237, *237*
dynamic host configuration protocol (DHCP) 56
dynamic web pages 66–7, 67t

E

elements and tags, HTML 130
email, links 142
embedded, content model 135, 136t
embedded systems 16–17
encryption 60, 74t
energy consumption 10
enhanced keyboard navigation, and semantic markup 252
environment, impact 115–17, 116t
error rate 48t
ethernet 49–50
ethical issues 121, 121–4t
evaluating IT systems 88–91
expansion capability 11
expression, moral/ethical issues 124t

F

features and functions 3–4, *3*
  connectivity 8–9, *9*, 15t
  and effects on design and use of a device 12, 12–15t
  energy consumption 10
  expansion capability 11
  media support 10
  performance 5–6, *5*
  portability 4–5, 18t
  security features 11–12
  storage 6–8, 13t
  user interface 7–8
fibre optic cables 39, *39*
files, size of 22–4, 22t
filter lists, JavaScript 214
firewalls 60, 75t
firmware 17
floating elements 183
flow
  content model 135, 135t
  web page design 228t
flowcharts 85–8, 85t, *86–8*, 91

focus and blur events, JavaScript 210–11
fonts 176–7, 225–7
form events, JavaScript 211, 211t
formats graphics/file 144, 144t
forms, web page 147–52, *148–51*, 149t, 179
free software 26t
functionality/usability testing 240
functions *see* features and functions

G

gateways 35t
global positioning system (GPS) 12t
grid layouts 184–6, *185–6*, 187–8, *188*
grouping records 109–10

H

hardware 2, 80
  see also features and functions
head elements 131–2
heading, content model 135, 136t
hidden attribute 133
horizontal scroll menus 236–7, *237*
HTML
  document structure 130–4
  page components 144–53, 144t, *145–51*, 153t
  structural markup 134–44, 135–6t, 136t, *137–9*, *141–2*
  see also CSS
hubs 33t
hyperlinks 131–2

I

ID attribute 133
images 144–5, 144t, 179, 188
inclusion, moral/ethical issues 122t
information, and data 94
infrared light 40–1, *41*, 53
inheritance 165
inline frames 153
input assistance 236t
input modalities 248t
inserting records into a database 106–7
intellectual property 119–20
interactive, content model 135, 136t
internet 64
internet protocol (IP) address 34t
internet of things (IoT) 15
IP addresses 56–7, 57t
IT systems
  dataflow 82–8, 83t, *84*, 85t, *86–8*
  designing and evaluating 88–91
  systems design 80–2, 81t
iteration, JavaScript 206–7

J

JavaScript
  Document Object Model (DOM) 195–6
  programmed functionality
    creating components using HTML/CSS/JavaScript 212–17, *217*
    error handling and debugging 217–19
    event triggers 208–11, 211t
    how to program functionality 200–8, 201t, 202t, 203t, 204t
  regular expressions 197–200
jitter 48t

K

key words 131
keyboard accessibility 233t
keyboard events, JavaScript 223
keys, database 99–100, 99t, 100t

L

language of the document 131, 133
latency 48t
legal issues 118–21
length values 175–6
licensing models, software 27t
light 40–1, *41*
links 131–2, 177–8
lists 140–2, *141–2*, 178, 214–15
live regions, and semantic markup 252
local area networks (LANs) 33t
location awareness 125

M

MAC addresses 56
machine learning 95
malware 59
measurement units 22–3, 22t
media access control (MAC) address 34t
media support 10
memory 13t
meta tags 131–2
microprocessors 16
microwave connections 40, *40*
miniaturisation 14t
mock-ups 224–5, *225*
modal boxes and images, JavaScript 213
modems 34t
monetisation, online communities 71–2t
moral issues 121, 121–4t
mouse events, JavaScript 210

N

national electrical code (NEC) 53
navigation 234t, 236–9, *237–9*, 252
network interface cards (NICs) 33t
network metrics 47, 47–8t
network protocols 53–5, 54t, 55t
network transmission media 37–41, *38–41*
networks 32
  components 32–7, 33–6t, *37*
  design 37–41, *38–41*, 56–7, 57t
  models and metrics , 42–9, 42t, *43–6*, 43t, 45t, 46t, 47–8t
  protocols
    communication 49–53, *52*
    network 53–5, 54t, 55t
  security 57–61, 58t
NFC 15t
Notepad++ 130, 155

O

object orientation, JavaScript 208
one and two-dimensional data structures 201–3, 201t, 202t, 203t
online communities 71–3, 71–2t, *72*
online environment
  the cloud 76–7, 76–7t
  internet/world wide web 64–7, *65*, 66t, 67t
  operating in the 68–75, *69–70*, 71–2t, *72*, 74–5t, *75*
open-source software 26t
operating systems 25–6
ordering records 109–10
OSI model 53, 54t
overlapping elements 183

P

packet loss 48t
passwords 58t, 74t
patches, software 28t
patents 119
pay-per-click advertising *72*, 72t
peer-to-peer networks 43–4, 43t
performance 5–6, *5*
  assessing 17–19, 17–18t
personal data, sharing 73–5, 74–5t, *75*
phrasing, content model 135, 136t
physical controls, security 60
pop-up boxes 215–16, 239, *239*
portability 4–5, 18t
positioning content 181–8, *181*, *185–6*, *188*
power efficiency 18t
powerline connections 39–40, *39*
predictability, accessibility 235t
privacy, moral/ethical issues 121t, 125
privacy settings 75t
processing data 19–22
processors 14t
proprietary software 26t
prototyping, web pages 226

Q

quick response (QR) code 15t

R

radio waves 40, *40*
readability, accessibility 235t
refresh 131
regular expressions, JavaScript 197–200
relationships, in a database 101–3
repeaters 36t
repetition, in JavaScript 205
responsive design 186–8, *188*
responsive images 188
RFID 14t
roles, and semantic markup 251
routers 34t

S

satellites 41, *41*
scalability 48t
screen size 184–6, *186*
scripts 131
search engine optimisation (SEO) 137, 247
search engines, and semantic code 246–7, *247*
sectioning, content model 135, 135t
security
  features 11–12
  network 57–61, 58t
  and the operating system 26
  and personal data 73–5, 74–5t, *75*, 118
security settings 59
seizures, and accessibility 233t
selecting data from a table 103–6
selection, in JavaScript 203–5, 204t
selectors 160–4
semantic markup 137, 139–40, *139*, 244–52, *245*, 245t, *247*, 247–50t
sensors 13t
server-side scripting 67, 67t
servers 36t
SIP 55
size of data files 22–4, 22t
slideshows, JavaScript 212–13
software 24–9, 26–7t, 28t, 80
speed
  of a device 17t
  network metric 47t
sponsored content 72t
states and properties, and semantic markup 252
static web pages 66–7, 67t
storage 6–8, 13t
structural markup 134–44, 135–6t, 136t, *137–9*, *141–2*
structured data 94–5
  in databases 98–100, 98t, 99t, 100t
structured query language (SQL) 103–12, 111t
style attribute 133
style guides, web page 225–6
styles 131, 176–9
sub-systems 82
subprocesses 87–8, *87–8*
subprograms, JavaScript 207–8
subscriptions, paid 72t
supercomputer 7
switches 34t
systems software 24

T

tabbed content, JavaScript 216–17, *217*
tabindex attribute 133
tables 145–7, 145t, *146–7*, 178–9
tags, HTML 130
targeted advertising 71t
TCP/IP model 54–5, 55t
technological convergence 15–16
tethering 45–6, *46*, 46t
text, styling 176–7
text alternatives, and accessibility 231t
throughput 47t
time, and accessibility 233t
time-based media 232t
touchscreens 13t
trade secrets 120
trademarks 119
transferring files, calculation for 23
transforms 190–2, *190*
transitions 188–90, *189–90*
twisted pair cables 38, *38*
two-factor authentication 58t
two/one-dimensional data structures 201–3, 201t, 202t, 203t
typography, web 227

U

unstructured data 95–7
updating software 28, 28t
upgrades, software 28t
usability, and web design 236–40, *237–9*
USB cables 9, *9*
user interface 7–8

V

validation, form 151–2
vertical menus 237, *237*
video and audio in web pages 152–3, 153t
viewports 132, 186–7
virtual private networks (VPNs) 74t
visual hierarchy 228t
VOIP 55

W

WAI-ARIA 251–2
Web Content Accessibility Guidelines (WCAG) 231
web pages
  audio and video 152–3, 153t
  colours 171–5, *174*
  designing
    principles of accessibility 231, 231–6t
    principles of design 222–30, 224–5, 228–300t
    principles of usability 236–40, 237–9
  forms 147–52, 148–51, 149t
  images 144–5, 144t
  inline frames 153
  tables 145–7, 145t, 146–7
'while' loops 205–6
Wi-Fi 50
wide area networks (WANs) 35t
wildcard searches 110–11
wireframes 223–4, *224*
wireless access 125–6, *126*
wireless access points (WAPs) 36t
working environments, online 68–9, *69*
world wide web 64–7, *65*, 66t, 67t
World Wide Web Consortium (W3C) 230, 240, 251

Z

ZigBee® 51

ACKNOWLEDGEMENTS

All Credits

Cover photo/illustration © MirageC/Moment/Getty images, SEAN GLADWELL/Moment/Getty images

Non-Prominent Image Credit(s):

123RF GB LIMITED: adam121 118, rawpixel 129, 221, Marina Basalaeva 229; **Alamy images:** Cavan Images 89; **Shutterstock:** Aleksandr Finch 246, alexdndz 225, Andrea Danti 60, Andrey_Popov 36c, 79, 94, 63, Antonio Guillem 51t, Argus 194, BestForBest 238, BritCats Studio 35t, chayanuphol 115, David Jancik 34t, dencg 34c, East pop 24, EgudinKa 128, elenabsl 126, 69, EPStudio20 114, Fractal Pictures 11, Gorodenkoff 2, Ground Picture 51b, Halfpoint 68, HL12 72, ivector 64, jamesteohart 1, jamie cross 158, Jiw Ingka 70, Joanna W Photo 49, junpiiiiiiiiiii 33b, Kateryna Reka 157, leo_photo 7, Leonardo da 154, Luca Lorenzelli 144, Magnetic Mcc 36b, maicasaa 243, Maksim Shmeljov 16, Marta Sher 65, McLittle Stock 222, metamorwork 31, 59, Morten B 117, Nebojsa Tatomirov 55, New Africa 36t, Ninetechno 35b, Ohmega1982 37, Panchenko Vladimir 134, Pixelbliss 47, Pong Ch 233, Primakov 239, PromKaz 33t, Rawpixel.com 15, ShlyahovaYulia 195, Song_about_summer 120, stefanphotozemun 223, stickasa 177, stockcreations 250, Summit Art Creations 8, Sunshine Seeds 34b, SvedOliver 6, Tee11 108, U-STUDIOGRAPHY DD59 226, vectorstudi 75, vectortatu 9, voronaman 80, ymgerman 50, Yurich 92, Zapp2Photo 95, Zita 242.